Essays on Machine Learning Methods for Data-Driven Marketing Decisions

Ryan Thomas Dew

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
under the Executive Committee
of the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2019

ABSTRACT

Essays on Machine Learning Methods for Data-Driven Marketing Decisions

Ryan Thomas Dew

Across three essays, I explore how modern statistical machine learning approaches can be used to glean novel marketing insights from data and to facilitate data-driven decision support in new domains. In particular, I draw on Bayesian nonparametrics, deep generative modeling, and modern Bayesian computational techniques to develop new methodologies that enhance standard marketing models, address modern challenges in data-driven marketing, and, as I show through applications to real world data, glean new, managerially relevant insights. Substantively, my work addresses issues in customer base analysis, the estimation of consumer preferences, and brand identity and logo design. In my first essay, I address how multi-product firms can understand and predict customer purchasing dynamics in the presence of partial information, by developing a Bayesian nonparametric model for customer purchasing activity. This framework yields an interpretable, model-based dashboard, which can be used to predict future activity, and guide managerial decision making. In my second essay, I explore the flexible modeling of customer brand choice dynamics using a novel form of heterogeneity, which I term dynamic heterogeneity. Specifically, I develop a novel doubly hierarchical Gaussian process framework to flexibly model how the preferences of individual customers evolve relative to one another over time, and illustrate the utility of the framework with an application to purchasing during the Great Recession. Finally, in my third essay, I explore how data and models can inform firms' aesthetic choices, in particular the design of their logos. To that end, I develop image processing algorithms and a deep generative model of brand identity

that links visual data with textual descriptions of firms and brand personality perceptions, which can be used for understanding design standards, ideation, and ultimately, data-driven design.

Contents

List of Tables

List of Figures

Acknowledgments

Foremost, I would like to acknowledge my loving and supportive partner Luke, as well as my family and friends, who enthusiastically supported my doctoral studies, and who kept me in high spirits all the way to the end. I would also like to acknowledge all of my academic mentors, especially my advisor Asim Ansari, from whom I learned it is normal to know the normal distribution, whose wisdom and wit were a huge boon during my time as a doctoral student, and without whose support and direction this dissertation would have been significantly worse. Thanks, too, to Olivier Toubia, whose creativity and insight have tremendouly helped in crafting the third essay of my dissertation, and to the rest of my committee—Oded Netzer, Kinshuk Jerath, and David Blei—whose feedback and guidance have shaped my thinking on these essays, and my approach to research generally. I would also like to acknowledge my classmates at Columbia, who have become great friends and colleagues, especially Khaled Boughanmi, Noah Castelo, and Yu Ding, with whom I had many inspirational, and many not-so-productive conversations. A special thanks to my mentor and now colleague, Pete Fader, who brought me into marketing academia, and encouraged me along the way. I would like to acknowledge the financial support of the INFORMS Society for Marketing Science, the Marketing Science Institute, and the Statistics in Marketing Section of the

American Statistical Association for generously funding this research. Finally, I acknowledge the love and affection of my beautiful new puppy, Momo, because of whom this dissertation took especially long to prepare.

I dedicate my dissertation to my loving and supportive partner, Luke.

Introduction

This dissertation consists of three essays, unified by the theme of fusing modern computational methods, in particular those from machine learning and Bayesian nonparametrics, with standard marketing models, frameworks, and ideas. Specifically, I focus on substantive issues in the domains of customer base analysis, the estimation of preferences, and the data-driven design of logos and brand identities. While these three domains may, at first glance, seem unrelated, I show through my research that, in each, new insights and better managerial decision support can result from an infusion of flexibile statistical modeling.

In my first essay, I address how multi-product firms can understand and predict customer purchasing dynamics in the presence of partial information. Customer purchasing activity is governed by both predictable customer-level effects, such as frequency, recency, and lifetime, as well as unpredictable and often unknown calendar time effects that could vary across products. Firms face the key challenge of automating the analysis process so that reliable insights can be gleaned without the need for repeated modeling adjustments across products. In this essay, I show how such automation can be achieved by modeling the different dynamic determinants of purchasing via latent functions that are estimated nonparametrically using Gaussian process priors. This yields a model-based

dashboard that can be used to understand variability in purchasing in calendar time, and characterize individual-level purchase propensities, which can then guide managerial actions.

In my second essay, I explore the flexible modeling of customer brand choice dynamics using a novel form of heterogeneity, which I term dynamic heterogeneity. Specifically, I develop a novel doubly hierarchical Gaussian process framework to flexibly model how the preferences of individual customers evolve relative to one another. I show how ignoring such heterogeneous preference evolution can distort inferences and mislead managers in customer targeting tasks, and illustrate the substantial gains in both model fit and insights by applying the specification to consumer packaged goods data from the era of the Great Recession.

Finally, in my third essay, I explore how data and models can inform firms' aesthetic choices, in particular the design of their logos. To that end, I develop image processing algorithms and a deep generative model of brand identity that links visual data with textual descriptions of firms and brand personality perceptions. In particular, I develop a multiview variational autoencoder to learn latent representations of brands, reflecting their brand identities, including visual components. When combined with my logo feature extraction algorithm, this multiview learning approach yields easily interpretable results that shed light on common design patterns, and can aid firms and designers in designing brand-relevant logos in a data-driven fashion.

Across these three essays, several themes emerge regarding the intersection of marketing and modern machine learning and Bayesian computational methods. The first theme is the importance of flexibly modeling dynamics: across both Essays 1 and 2, I demonstrate how relaxing standard assumptions about time dynamics via Bayesian nonparametrics yields better model performance and deeper insights. In

both of these cases, the methodological innovation comes from recasting standard probability models as problems of estimating unknown functions, then performing inference on those functions. Bayesian nonparametrics and machine learning have given us a plethora of ways of estimating unknown functions. I have shown two ways in which such an approach adds flexibility, and improves insights and decisions, and believe that adding flexibility to marketing models in this fashion will likely be a fundamental way in which these methods enhance marketing models, and thus, marketing decisions going forward. While not related to dynamics, my third essay again leverages the same idea of flexible, probabilistic modeling, by drawing on a modern method of learning a joint distribution across different data modalities: the multiview variational autoencoder. In this case, flexibility is afforded by linking representations and data through neural networks, which again can be seen a recasting of a classical problem (dimensionality reduction) in terms of function estimation.

Another common theme is the utility of modern approaches to Bayesian inference, in particular methods based on gradients and automatic differentiation, including Hamiltonian Monte Carlo and no-U-turn sampling methods, and black box variational inference. These techniques, with their implementations in modern probabilistic programming languages, can allow marketers to focus on modeling, rather than inference, and to make the most reasonable assumptions about the data, rather than those that are computationally convenient. These techniques were used in all three essays of my dissertation, thus showcasing another way in which computational methods can enhance the capacity of marketing managers to make data-driven decisions.

Finally, as illustrated by the final essay of my dissertation, machine learning and image processing techniques allow us to leverage new types of data, particularly

visual data. This essay is among the first papers in marketing to directly leverage image data, despite the prevalence of visual information in marketing. Aesthetics is fundamental to advertising, branding, product design, packaging, e-commerce, and a huge number of other domains in marketing. As illustrated in my third essay, directly using image and other unstructured data in marketing models, and developing and implementing new models that can handle such data, is another way in which machine learning methods can enhance the ability of marketers to make data-driven decisions.

1

Bayesian Nonparametric Customer Base Analysis with Model-based Dashboards

A paper based on this essay was published in Marketing Science, Vol. 37 (2018), under the title "Bayesian Nonparametric Customer Base Analysis with Model-based Visualizations." That paper is jointly authored with Asim Ansari.

Abstract

Marketing managers are responsible for understanding and predicting customer purchasing activity, a task that is complicated by a lack of knowledge of all of the calendar time events that influence purchase timing. Yet, isolating calendar time variability from the natural ebb and flow of purchasing is important, both for accurately assessing the influence of calendar time shocks to the spending process, and for uncovering the customer-level patterns of purchasing that robustly predict future spending. A comprehensive understanding of purchasing dynamics therefore requires a model that flexibly integrates both known and unknown calendar time determinants of purchasing with individual-level predictors such as interpurchase time, customer lifetime, and number of past purchases. In this paper, we develop a Bayesian nonparametric framework based on Gaussian process priors, which integrates these two sets of predictors by modeling both through latent functions that jointly determine purchase propensity. The estimates of these latent functions yield a visual representation of purchasing dynamics, which we call the model-based dashboard, that provides a nuanced decomposition of spending patterns. We show the utility of this framework through an application to purchasing in free-to-play mobile video games. Moreover, we show that in forecasting future spending, our model outperforms existing benchmarks.

1.1 Introduction

Marketers in multi-product companies face the daunting task of understanding the ebb and flow of aggregate sales within and across many distinct customer bases. Such spending dynamics stem from both the natural stochastic process of purchasing that is characterized by customers' interpurchase times, lifetimes with the firm, and number of past purchases, and from the influence of managerial actions and shocks operating in calendar time. These other shocks are often outside the control of the company, and include events such as holidays, barriers to purchasing like website outages, and competitor actions. While individual-level factors such as the recency of purchasing are often powerful predictors of future spend activity, managers think and act in calendar time. Hence, to successfully execute a customer-centric marketing strategy, managers need to understand how calender time events interact with individual-level effects in generating aggregate sales.

An accurate accounting of the underlying drivers of spending is not possible unless both individual-level and calendar time effects are simultaneously modeled. For example, in models of spending that omit calendar time and rely solely on individual-level effects, momentary disruptions in spending that occur in calendar time may be erroneously conflated with predictable, individual-level purchase propensities. Similarly, a small bump in spending on any given calendar day could represent random noise if many customers are still active on that day, or a significant calendar time event if few customers are still active. Importantly, activity level is unobserved, but can be captured by individual-level variables like interpurchase time. Flexibly including both types of effects in an individual-level model of purchase propensity is thus crucial for dynamic customer base analysis, and the development of such a framework is our primary objective.

In this paper, we describe a flexible and robust Bayesian nonparametric framework for customer base analysis that accomplishes that objective by probabilistically modeling purchase propensities in terms of underlying dynamic components. We demonstrate the utility of our new framework on spending data from mobile video games. Our model uses Gaussian process priors over latent functions to integrate events that occur at multiple time scales and across different levels of aggregation, including both calendar time and individual-level time scales like interpurchase time, time since first purchase (customer lifetime), and number of past purchases. Its nonparametric specification allows for the flexible modeling of different patterns of effects, such that the model can be seamlessly applied across different customer bases and dynamic contexts. The resulting latent function estimates facilitate automatic model-based visualization and prediction of spending dynamics.

Customer base analysis is central to modern marketing analytics. Contributions in this area have focused on the stochastic modeling of individuals in terms of interpurchase time and lifetime, in contractual and non-contractual settings (Fader et al., 2005; Schmittlein et al., 1987; Fader et al., 2010; Schweidel and Knox, 2013). These papers show that customer-level effects can explain much of the variability of spending over time. However, they typically omit, or assume a priori known, calendar time effects. Events in calendar time, including marketing efforts and exogenous events such as competitor actions, holidays, and day-of-the-week effects, can substantially impact spending in many industries. For digital products, such as those in our application, relevant calendar events include product changes that are launched simultaneously to all customers, and exogenous shocks such as website or e-commerce platform outages and crashes. Moreover, many of these events pose a common problem to marketing analysts: although calendar time events undoubtedly influence spend rates, analysts may be unaware of

the form of that influence, or of the very existence of certain events. This problem is exacerbated in larger companies, where the teams responsible for implementing marketing campaigns or managing products may be distinct from the analytics team, and where information may not flow easily across different organizational silos.

To cope both with such information asymmetries and with unpredictable dynamics in spending, sophisticated managers often rely on aggregate data methods, including exploratory data analyses, statistical process control, time series models (Hanssens et al., 2001), and predictive data mining methods (Neslin et al., 2006). These tools can forecast sales, model the impact of calendar time events, and provide metrics and visual depictions of dynamic patterns that are easy to grasp. Unfortunately, these methods typically ignore individual-level predictors of spend, like those captured by customer base analysis models, which precludes their use in characterizing customer-level spend behaviors and in performing CRM-relevant tasks. Furthermore, not including these individual-level effects means these models cannot account for the latent activity level of customers, which may in turn lead to an inaccurate understanding of the true nature of calendar time events.

Building on both the customer base analysis and aggregate data approaches, we use Bayesian nonparametric Gaussian process (GP) priors to fuse together latent functions that operate both over calendar time and over more traditional individual-level inputs, such as interpurchase time, customer lifetime, and purchase number. In this way, we integrate calendar time insights into the customer base analysis framework. We use these latent functions within a discrete hazard specification to dynamically model customer purchase propensities, while controlling for unobserved heterogeneity. We term the resulting model the Gaussian Process Propensity Model (GPPM). While Bayesian nonparametrics have been

successfully applied to marketing problems (e.g. Ansari and Mela, 2003; Wedel and Zhang, 2004; Kim et al., 2007; Rossi, 2013; Li and Ansari, 2014), to the best of our knowledge, our paper is the first in marketing to take advantage of the powerful GP methodology. It is important to note that, although our paper applies GPs in the context of customer purchasing, GPs provide a general mechanism for estimating latent functions, and can be employed in many other substantive contexts. We therefore also provide an accessible introduction to GPs in general, to encourage their wider adoption within marketing.

In our application, the GP nonparametric framework means that the shapes of the latent propensity functions that govern purchasing are automatically inferred from the data, thus providing the flexibility to robustly adapt to different settings, and to capture time-varying effects, even when all the information about inputs may not be available. The inferred latent functions allow a visual representation of both calendar time and individual-level patterns that characterize spend dynamics, something that is not possible in standard probability models, where the output is often a set of possibly unintuitive parameters. We refer to the collection of these plots as the model-based dashboard, as it gives a visual summary of the patterns of spending in a particular customer base, and serves as a tool for analyzing the spending dynamics within and across customer bases. It is important to note that these model-based dashboards are distinct from real-time dashboards that continuously stream various marketing metrics, like those described in Pauwels et al. (2009).

In this paper, we begin by describing what Gaussian process priors are (Section 2.1), and how they can be used to specify latent dynamics in a model for dynamic customer base analysis (Sections 2.2 and 2.3). We then apply our model to spending data from two mobile video games owned by a large American video game

publisher. These games are quite distinct, spanning different content genres and target audiences. We show how the parameter estimates and accompanying model-based dashboards generated from our approach can facilitate managerial understanding of the key dynamics within each customer base, both in the aggregate and at the individual level (Sections 3.1 and 3.2). We compare the GPPM to benchmark probability models, including different buy-till-you-die variants such as the BGNBD (Fader et al., 2005) and the Pareto-NBD (Schmittlein et al., 1987), hazard models with and without time-varying covariates (e.g. Gupta, 1991; Seetharaman and Chintagunta, 2003), and variants of the discrete hazard approach, including a sophisticated state-space specification, and show that the GPPM significantly outperforms these existing benchmarks in fit and forecasting tasks (Section 3.4). We conclude by summarizing the benefits of our framework, citing its limitations, and identifying areas of future research.

1.2 Modeling Framework

In our framework for dynamic customer base analysis, we focus on flexibly modeling individual-level purchase propensity. We model this latent propensity in terms of the natural variability in purchase incidence data along four dimensions: calendar time, interpurchase time (recency), customer lifetime, and number of past purchases. Our focus on modeling purchase incidence is consistent with the majority of the literature on customer base analysis, and also fits nicely with our application area, where we focus on purchasing of a single product, and where there is minimal variability in spend amount.[1] We use a discrete-time hazard framework to specify the purchase propensity, as most customer-level data are available at a

[1]Throughout the rest of the paper, we use the words purchasing and spending interchangeably to refer specifically to purchase incidence.

discrete level of aggregation. This is also the case in our application, where daily data are available.

The observations in our data consist of a binary indicator y_{ij} that specifies whether customer i made a purchase at observation j, and a corresponding tuple $(t_{ij}, r_{ij}, \ell_{ij}, q_{ij})$ containing the calendar time, recency, customer lifetime, and number of past purchases, respectively. Recency here refers to interpurchase time, or the time since the customer's previous purchase, while customer lifetime refers to the time since the customer's first purchase. Depending on the context, a vector $\mathbf{z}_i$ of demographics or other time invariant variables, such as the customer acquisition channel or acquisition date, may also be available. The probability of customer i purchasing is modeled as

$$\Pr(y_{ij} = 1) = \text{logit}^{-1}\left[\alpha(t_{ij}, r_{ij}, \ell_{ij}, q_{ij}) + \mathbf{z}_i'\boldsymbol{\gamma} + \delta_i\right], \tag{1.1}$$

where, $\text{logit}^{-1}(x) = \frac{1}{1+\exp(-x)}$. We see in Equation 1.1 that the purchasing rate is driven by a time-varying component $\alpha(.)$ and two time invariant effects, $\mathbf{z}_i'\boldsymbol{\gamma}$ and δ_i, which capture the observed and unobserved sources of heterogeneity in base spending rates, respectively. This setup models spend dynamics via aggregate trajectories—that is, all customers are assumed to follow the same dynamic pattern—while maintaining individual heterogeneity in the spending process via the random effect δ_i and by using other observed individual-specific variables, $\mathbf{z}_i$, when available. In our application, we will focus exclusively on unobserved heterogeneity. It is important to note that while calendar time is an aggregate time scale, the recency, lifetime, and purchase number dimensions are individual-level time scales. That is, customers may, at any given point in calendar time t, be at a different positions in the $(r_{ij}, \ell_{ij}, q_{ij})$ subspace, and therefore the aggregate sales at any given calendar time t are the amalgam of the activities of customers who differ widely in

their expected purchase behaviors.

The heart of our framework involves the specification of the purchase propensity, $\alpha(t_{ij}, r_{ij}, \ell_{ij}, q_{ij})$. We treat $\alpha(.)$ as a latent function and model it nonparametrically using Gaussian process priors (Rasmussen and Williams, 2006; Roberts et al., 2013). The nonparametric approach models random functions flexibly and allows us to automatically accommodate different patterns of spend dynamics that may underlie a given customer base. These dynamics operate along all four of our dimensions. Furthermore, these dynamics may operate at different time scales within a single dimension, including smooth long-run trends and short-term patterns, as well as cyclic variation, which are inferred from the data. To allow such rich structure, we use an additive combination of unidimensional GPs to specify and estimate the multivariate function $\alpha(t_{ij}, r_{ij}, \ell_{ij}, q_{ij})$.

1.2.1 Gaussian Process Priors

We begin by describing GPs and highlight how they can nonparametrically capture rich, dynamic patterns in a Bayesian probability model. A Gaussian process is a stochastic process $\{f(\tau) : \tau \in \mathcal{T}\}$ indexed by input elements τ such that, for any finite set of input values, $\boldsymbol{\tau} = \{\tau_1, \tau_2, \ldots, \tau_M\}$, the corresponding set of function outputs, $f(\boldsymbol{\tau}) = \{f(\tau_1), f(\tau_2), \ldots, f(\tau_M)\}$, follows a multivariate Gaussian distribution. The characteristics of the stochastic process are defined by a mean function and a covariance function, also called a kernel. For a fixed set of inputs, a Gaussian Process reduces to the familiar multivariate Gaussian distribution, with a mean vector determined by the GP's mean function, and a covariance matrix determined by its kernel. However, unlike a standard multivariate normal distribution that is defined over vectors of fixed length, a Gaussian process defines a

distribution over outputs for any possible set of inputs. From a Bayesian perspective, this provides a natural mechanism for probabilistically specifying uncertainty over functions. Since the estimated function values are the parameters of a GP, the number of parameters grows with the number of unique inputs, making the model nonparametric.

While GPs are often defined over multidimensional inputs, for simplicity of exposition, we begin by assuming a unidimensional input, $\tau \in \mathbb{R}$ (e.g., time). To fix notation, suppose f is a function that depends on that input. Let $\boldsymbol{\tau}$ be a vector of M input points, and let $f(\boldsymbol{\tau})$ be the corresponding vector of output function values. As described above, a GP prior over f is completely specified by a mean function, $m(\tau) = \mathrm{E}[f(\tau)]$, and a kernel, $k(\tau, \tau') = \mathrm{Cov}[f(\tau), f(\tau')]$, that defines a positive semidefinite covariance matrix

$$K(\boldsymbol{\tau}, \boldsymbol{\tau}) = \begin{pmatrix} k(\tau_1, \tau_1) & k(\tau_1, \tau_2) & \dots & k(\tau_1, \tau_M) \\ k(\tau_2, \tau_1) & k(\tau_2, \tau_2) & \dots & k(\tau_2, \tau_M) \\ \vdots & \vdots & \ddots & \vdots \\ k(\tau_M, \tau_1) & k(\tau_M, \tau_2) & \dots & k(\tau_M, \tau_M) \end{pmatrix}, \tag{1.2}$$

over all the outputs. We discuss specific forms of the mean function and kernel in Sections 2.1.1 and 2.1.2. Generally, these functions are governed by a small set of hyperparameters that embody certain traits of the GP. For instance, the squared exponential kernel, which we discuss in considerable detail in Section 2.2.2, is given by $k_{\mathrm{SE}}(\tau_i, \tau_j) = \eta^2 \exp\{-(\tau_i - \tau_j)^2/(2\rho^2)\}$. This form encodes the idea that nearby inputs should have related outputs through two hyperparameters: an amplitude, η, and a smoothness, ρ. Intuitively, these two hyperparameters determine the traits of the function space being modeled by a GP with this kernel.

Given a fixed vector of inputs $\boldsymbol{\tau}$, letting $f(\boldsymbol{\tau}) \sim \mathcal{GP}(m(\boldsymbol{\tau}), k(\boldsymbol{\tau}, \boldsymbol{\tau}'))$ is

equivalent to modeling the vector of function outputs via a marginal multivariate Gaussian $f(\boldsymbol{\tau}) \sim \mathcal{N}(m(\boldsymbol{\tau}), K(\boldsymbol{\tau}, \boldsymbol{\tau}))$. The mean $m(\boldsymbol{\tau})$ and covariance matrix $K(\boldsymbol{\tau}, \boldsymbol{\tau})$ of the above multivariate normal marginal distribution are again parsimoniously determined through the small set of hyperparameters underlying the mean function and kernel of the GP. The fact that the marginal of a GP is a multivariate normal distribution makes it easy to comprehend how function interpolation and extrapolation work in this framework. Conditioned on an estimate for the function values at the observed inputs, and on the mean function and kernel hyperparameters, the output values for the latent function f for some new input points $\boldsymbol{\tau}^*$ can be predicted using the conditional distribution of a multivariate normal. Specifically, the joint distribution of the old and new function values is given by

$$\begin{bmatrix} f(\boldsymbol{\tau}) \\ f(\boldsymbol{\tau}^*) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m(\boldsymbol{\tau}) \\ m(\boldsymbol{\tau}^*) \end{bmatrix}, \begin{bmatrix} K(\boldsymbol{\tau}, \boldsymbol{\tau}) & K(\boldsymbol{\tau}, \boldsymbol{\tau}^*) \\ K(\boldsymbol{\tau}^*, \boldsymbol{\tau}) & K(\boldsymbol{\tau}^*, \boldsymbol{\tau}^*) \end{bmatrix} \right), \tag{1.3}$$

and hence the conditional distribution of the new outputs can be written as

$$\begin{aligned} f(\boldsymbol{\tau}^*) \sim \mathcal{N}(&m(\boldsymbol{\tau}^*) + K(\boldsymbol{\tau}^*, \boldsymbol{\tau}) K(\boldsymbol{\tau}, \boldsymbol{\tau})^{-1} [f(\boldsymbol{\tau}) - m(\boldsymbol{\tau})], \\ &K(\boldsymbol{\tau}^*, \boldsymbol{\tau}^*) - K(\boldsymbol{\tau}^*, \boldsymbol{\tau}) K(\boldsymbol{\tau}, \boldsymbol{\tau})^{-1} K(\boldsymbol{\tau}, \boldsymbol{\tau}^*)). \end{aligned} \tag{1.4}$$

This equation again makes clear that the kernel and mean functions determine the distribution of the output values both for existing and new inputs. As the mean and covariance of the marginal multivariate normal are parametrized via the mean and kernel functions, the GP remains parsimonious, and can interpolate and extrapolate seamlessly for any set of input values. The choice of mean function allows us to model different a priori expected functional forms, while the kernel determines how much the functions deviate nonparametrically from that mean function.

Mean Functions

The mean function captures expected functional behaviors. Within the range of observed inputs, the mean function often has very little influence over the estimated function values; instead, the properties of the estimated function are largely determined by the kernel, as we describe in the next section. Because of this, in many GP applications, the mean function is set to a constant, reflecting no prior assumptions about functional form. However, far from the range of observed inputs, the posterior expected function values revert to the mean function.[2] In some applications, this mean reverting behavior in combination with a constant mean function is problematic, as we may expect the function values to be increasing or decreasing, both in and out of the range of inputs. To capture this expected behavior, we may choose to use a non-constant mean function.

In this paper, we use either a constant mean function, or a parametric monotonic power mean function, given by $m(\tau) = \lambda_1(\tau - 1)^{\lambda_2},\ \lambda_2 > 0$. This specification captures expected monotonic behavior, while also allowing for a decreasing marginal effect over the input.[3] We use $(\tau - 1)$ and restrict $\lambda_2 > 0$, to be consistent with our identification restrictions that we describe later. We emphasize again that the mean function sets an expectation over function values, but does not restrict them significantly. The GP structure allows functions to nonparametrically deviate from the mean function, resulting in function estimates that differ from the

[2]This behavior can be seen through Equation 1.4, in conjunction with, for example, the squared exponential kernel, briefly mentioned above, which has functional form $k_{\text{SE}}(\tau_i, \tau_j) = \eta^2 \exp\{-(\tau_i - \tau_j)^2/(2\rho^2)\}$. As the distance between the observed inputs and the new input grows, the value of the kernel goes to zero, and we see the mean in Equation 1.4 will revert to the mean function. This mean reverting property is dependent on the kernel being stationary, meaning that it depends only on the distance between inputs. We refer the interested reader to Rasmussen and Williams (2006), for a comprehensive discussion of these issues.

[3]We note that the properties of this specification are suitable for our specific application, but may not be suitable in other domains and substantive applications.

mean's parametric form. This is obvious in all panels of Figure 1.1, where we plot random draws from GPs with different mean functions and kernels. Across the panels of Figure 1.1, we see shapes that are sometimes dramatically different from the respective constant and power mean functions that generated them. The main role of the mean function is in extrapolating far from the range of the observed inputs, where it determines expected function behavior in the absence of data. While we use only these two mean functions as a simple way of capturing our prior expections, any parametric form could be potentially used as a mean function. Given the capacity of the GP to capture deviations from parametric forms, it is generally considered best practice to use simple mean functions, and let the GP capture any complexities.

Kernels

The kernel defines much of the fundamental structure of a GP, and in combination with the mean function, determines the latent function space of a GP prior. As such, kernels are the primary source of model specification when working with GP priors. Any function over two inputs that results in a positive semidefinite gram matrix can be used as a kernel, and many different kernel forms have been explored in the GP literature (Rasmussen and Williams, 2006, Chapter 4). Kernels encode the structure of functions via a small number of hyperparameters, leading to highly flexible yet parsimonious model specification. In this paper, we use two simple kernels that are suitable building blocks for describing functions in our context.

The first kernel is the squared exponential kernel (SE) defined as

$$k_{\text{SE}}(\tau_j, \tau_k; \eta, \rho) = \eta^2 \exp\left\{-\frac{(\tau_j - \tau_k)^2}{2\rho^2}\right\}, \tag{1.5}$$

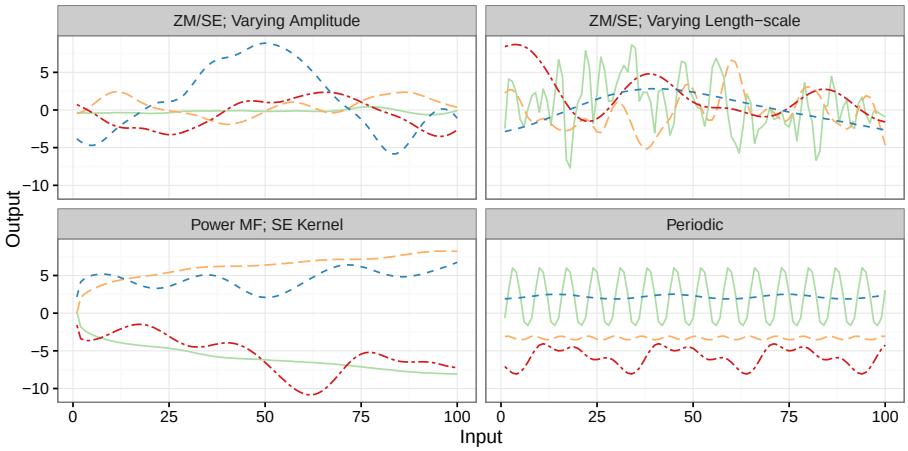

Figure 1.1: Examples of mean function/kernel combinations. Top-left: zero mean function and SE kernel with $\rho^2 = 50$ and $\eta^2 \in \{0.1, 1, 5, 20\}$; Top-right: zero mean function and SE kernel with $\rho^2 \in \{1, 10, 100, 1000\}$; Bottom-left: power mean function $m(\tau) = \pm 2(\tau - 1)^{0.3}$ and SE kernel with $\rho^2 = 100$ and $\eta^2 \in \{0.1, 5\}$; Bottom-right: periodic kernels with $\eta^2 = 10$, $\rho^2 \in \{2, 100\}$, and $\omega \in \{7, 30\}$.

where the hyperparameter $\eta > 0$ is the amplitude, and $\rho > 0$ is the characteristic length-scale or "smoothness." The amplitude can be best explained by considering the case when $\tau_j = \tau_k \equiv \tau$. In this case, $k(\tau, \tau) = \eta^2$, which is the variance of the normal distribution at the fixed input value τ. More generally, η^2 captures variance around the mean function. If $\eta \to 0$, the GP will largely mirror its mean function. We illustrate this using both the constant and power mean functions in the left column of Figure 1.1, where we randomly draw GPs with a fixed ρ and varying η values. From these two panels, we can see that small values of η, as in the light-colored solid (green) and long-dash (yellow) curves, yield functions that stay closer to their mean functions, relative to the dark-colored dot-dash (red) and short-dash (blue) curves with higher η values. The characteristic length-scale ρ intuitively indicates how far apart two input points need to be for the corresponding outputs to be uncorrelated. Hence, a high value of ρ corresponds to very smooth functions, while a small value of ρ yields jagged, unpredictable functions. We see this illustrated in the top-right panel of Figure 1.1, where we fix the amplitude η and vary the length-scale ρ. We can see a clear contrast between the highly jagged

solid (green) curve with $\rho^2 = 1$, and the increasingly smooth dashed curves, with $\rho^2 \in \{10, 100, 1000\}$.

The second kernel we use is the periodic kernel, defined by

$$k_{\text{Per}}(\tau_j, \tau_k; \omega, \eta, \rho) = \eta^2 \exp\left\{-\frac{\sin^2\left(\pi(\tau_j - \tau_k)^2/\omega\right)}{\rho^2}\right\}. \tag{1.6}$$

This kernel allows for periodic functions with period ω that are again defined by an amplitude η and a length-scale ρ. Note that this type of variability could also be captured by the squared exponential kernel; the benefit of using the periodic kernel is that forecasts based on this kernel will always precisely mirror the estimated pattern. Hence, any predictable cyclic variability in the data would be captured both in and out-of-sample. In the bottom-right panel of Figure 1.1, we plot four draws from different periodic kernels. There, we show different cycle lengths (30 days and 7 days), together with differing smoothness and amplitude parameters.

In addition to the above described kernels, many other types have been proposed in the GP literature. In this paper, we use the simplest kernels that exemplify a given trait (stationary variability with the SE and cyclicality with the periodic). These are by far the most commonly used kernels, the squared exponential especially serving as the workhorse kernel for the bulk of the GP literature. Additional kernels include the rational quadratic, which can be derived as an infinite mixture of squared exponential kernels, and the large class of Matern kernels, which can capture different levels of differentiability in function draws.

Additivity

Just as the sum of Gaussian variates is distributed Gaussian, the sum of GPs is also a GP, with a mean function equal to the sum of the mean functions of the component GPs, and its kernel equal to the sum of the constituent kernels. This is called the additivity property of GPs, and can allow us to define a rich structure even along a single dimensional input. Specifically, the additivity property allows us to model the latent function f as a sum of sub-functions on the same input space, $f(\tau) = f_1(\tau) + f_2(\tau) + \ldots + f_J(\tau)$, where each of these sub-functions can have its own mean function, $m_j(\tau)$, and kernel, $k_j(\tau, \tau')$. The mean function and kernel of the function f are then given by $m(\tau) = \sum_{j=1}^{J} m_j(\tau)$ and $k(\tau, \tau') = \sum_{j=1}^{J} k_j(\tau, \tau')$, respectively. This allows us to flexibly represent complex patterns of dynamics even when using simple kernels like the squared exponential. We can, for example, allow the different sub-functions to have different squared exponential kernels that capture variability along different length-scales, or add a periodic kernel to isolate predictable cyclic variability of a given cycle length. It is through this additive mechanism that we represent long-run and short-run variability in a given dimension, for instance, or isolate predictable periodic effects from unpredictable noise, as we discuss in Section 1.2.2.[4] Until now, we have focused on illustrating GPs in unidimensional contexts. We now show how additivity can be leveraged to construct GPs for multidimensional functions.

[4]In general, determining the number of additive components suitable for a given application requires both substantive knowledge and expectations about the nature of the dynamics at work, and data-driven evidence from the estimated hyperparameter values. For instance, depending on the kernel, a small amplitude hyperparameter compared to the output scale could indicate the component is relatively uninfluential in describing the results. Similarly, if the length-scale is estimated to be very large, this can indicate minimal dynamics are being uncovered by that component. Both of these phenomena can indicate redundancy in the specification. Kernel specification is a rich topic in the GP literature, and the interested reader can find considerable discussion in Rasmussen and Williams (2006), Chapter 5.

Multidimensional GPs

In practice, we are often interested in estimating a multidimensional function, such as the $\alpha(.)$ function in Equation 1.1. Let $h(.)$ be a generic multidimensional function from $\mathbb{R}^D$ to $\mathbb{R}$. The inputs to such a function are vectors of the form $\boldsymbol{\tau}_m \equiv (\tau_m^{(1)}, \tau_m^{(2)}, \ldots, \tau_m^{(D)}) \in \mathbb{R}^D$, for $m = 1, \ldots, M$, such that the set of all inputs is an $M \times D$ matrix. Just as before, $h(.)$ can also be modeled via a GP prior. While there are many ways in which multi-input functions can be modeled via GPs, a simple yet powerful approach is to consider $h(.)$ as a sum of single input functions, $h_1(.), h_2(.), \ldots, h_D(.)$, and model each of these unidimensional functions as a unidimensional GP with its own mean function and kernel structure (Duvenaud et al., 2013). The additivity property implies that additively combining a set of unidimensional GP's over each dimension of the function is equivalent to using a particular sum kernel GP on the whole, multidimensional function. We use such an additive structure to model $\alpha(t_{ij}, r_{ij}, \ell_{ij}, q_{ij})$ in the GPPM.

Additively separable GPs offer many benefits: first, they allows us to easily understand patterns along a given dimension, and they facilitate visualization, as the sub-functions are unidimensional. Second, the additivity property implies that the combined stochastic process is also a GP. Finally, the separable structure reduces computational complexity. Estimating a GP involves inverting its kernel matrix. This inversion requires $O(M^3)$ computational time and $O(M^2)$ storage demands for M inputs. In our case, as the inputs $(t_{ij}, r_{ij}, \ell_{ij}, q_{ij})$ can only exist on a grid of fixed values, we will have $L < M$ inputs, where L corresponds to all unique observed $(t_{ij}, r_{ij}, \ell_{ij}, q_{ij})$ combinations. Despite the reduction, this is a very large number of inputs, and would result in considerable computational complexity, without the separable structure. The additive specification reduces this computational burden to that of inverting multiple (in our case, six) $T \times T$

matrices, where $T \ll M$ is the number of time periods observed in the data.

Comparison to Other Function Estimation Methods

As Gaussian process priors are new to marketing, it is worthwhile to briefly summarize the rationale for using them, instead of other flexible methods for modeling latent functions like simple fixed effects, splines, or state space models. Foremost, GPs allow for a structured decomposition of a single process into several subprocesses via the additivity property. This additive formulation facilitates a rich representation of a dynamic process via a series of kernels that can capture patterns of different forms (e.g., periodic vs. non-periodic) and operate at different time scales. Yet, as the sum of GPs is a GP, the specification remains identified, with a particular mean and covariance kernel. Achieving a similar representation with other methods is either infeasible or more difficult.[5] Moreover, GPs are relatively parsimonious, and when estimated in a Bayesian framework, tend to avoid overfitting. Bayesian estimation of GPs involves estimating the function values and hyperparameters jointly, thus determining both the traits of the function, and the function values themselves. As the flexibility of the latent functions is controlled via a small number of hyperparameters, we retain parsimony. Moreover, the structure of the marginal likelihood of GPs, obtained by integrating out the function values, clearly shows how the model makes an implicit fit versus complexity tradeoff whereby function flexibility, as captured by the hyperparameters, is balanced by a penalty that results in the regularization of the fit (for details, see Rasmussen and

[5]While we emphasize the relative benefits of GP priors here, we also note that there are many links between these methods, including between GP methods and smoothing splines (Kalyanam and Shively (1998) and Shively et al. (2000)), and between GP methods and state space models. We include a sophisticated state space analog of our model in our benchmarks. Our state space formulation is also closely related to cubic spline specifications (see Durbin and Koopman (2012) for details). As we will describe later, although this method produces fits that are roughly on par with the GP approach, we cannot easily obtain the decompositions that are natural in the GP setting.

Williams (2006), Section 5.4.1).

1.2.2 Full Model Specification

The flexibility afforded by GP priors makes them especially appropriate for modeling our latent, time-varying function, $\alpha(t_{ij}, r_{ij}, \ell_{ij}, q_{ij})$. Recall that the basic form of the GPPM is:

$$\Pr(y_{ij} = 1) = \text{logit}^{-1}\left[\alpha(t_{ij}, r_{ij}, \ell_{ij}, q_{ij}) + \mathbf{z}_i'\boldsymbol{\gamma} + \delta_i\right]. \tag{1.7}$$

For ease of exposition, we will subsequently omit the ij subscripts. For simplicity and to reduce computational complexity, we assume an additive structure,

$$\alpha(t, r, \ell, q) = \alpha_{\text{T}}(t) + \alpha_{\text{R}}(r) + \alpha_{\text{L}}(\ell) + \alpha_{\text{Q}}(q), \tag{1.8}$$

and model each of these functions using separate GP priors. This structure and the nonlinear nature of the model implies an interaction between the effects: for example, if the recency effect is very negative, calendar time events can do little to alter the spend probability. While additivity is a simplifying assumption, in our application, this compensatory structure seems to explain the data well.

To specify each of these additive components, we return to the mean functions and kernels outlined in Sections 2.1.1 and 2.1.2, and to the additivity property of GPs from Section 2.1.3. Recall that the mean function encodes the expected functional behavior: with the constant mean function, we impose no expectations; with the power mean function, we encode expected monotonicity. The kernel choice endows the GP with additional properties: a single SE kernel allows flexible variation with one characteristic length-scale, while the periodic kernel

allows the GP to exhibit predictable cyclic behavior with a given period. Additivity allows us to combine these kernel properties, to achieve variation along more than one length-scale, or to isolate predictable cyclic behavior in a given dimension. We can use these general traits of mean function and kernel combinations to specify our model, based on the expected nature of the variation along a given dimenion. Below, we explain the specification used in our application. The GPPM framework is highly flexible, and throughout the following sections, we also explain how this specification can be modified to handle more general settings.

Calendar Time In calendar time, we expect two effects to operate: long run trends, and short run disturbances. These short run events could include promotions, holidays, or other shocks to the purchasing process. Furthermore, we expect cyclicality such that purchasing could be higher on weekends than on weekdays, or in particular months or seasons. As we describe later, in our application, given the span of our data, we expect only one periodic day of the week (DoW) effect. Together, this description of spend dynamics implies a decomposition of α_{T} into three sub-components,

$$\alpha_{\mathrm{T}}(t) = \alpha_{\mathrm{T}}^{\mathrm{Long}}(t) + \alpha_{\mathrm{T}}^{\mathrm{Short}}(t) + \alpha_{\mathrm{T}}^{\mathrm{DoW}}(t), \tag{1.9}$$

where we model each component such that,

$$\begin{aligned}
\alpha_{\mathrm{T}}^{\mathrm{Long}}(t) &\sim \mathcal{GP}(\mu, k_{\mathrm{SE}}(t, t'; \eta_{\mathrm{TL}}, \rho_{\mathrm{TL}})), \\
\alpha_{\mathrm{T}}^{\mathrm{Short}}(t) &\sim \mathcal{GP}(0, k_{\mathrm{SE}}(t, t'; \eta_{\mathrm{TS}}, \rho_{\mathrm{TS}})), \\
\alpha_{\mathrm{T}}^{\mathrm{DoW}}(t) &\sim \mathcal{GP}(0, k_{\mathrm{Per}}(t, t'; \omega = 7, \eta_{\mathrm{TW}}, \rho_{\mathrm{TW}})).
\end{aligned}$$

Without loss of generality, we impose $\rho_{\text{TL}} > \rho_{\text{TS}}$, to ensure that the long-run component captures smoother variation than the short-run component. We use constant mean functions here because, a priori, we do not wish to impose any assumptions about calendar time behavior. The constant mean μ in the long-run component captures the base spending rate in the model. Far from the range of the data, this specification implies the posterior mean of these effects will revert to this base spending rate, reflecting our lack of a priori knowledge about these effects.

This specification is very general, and has shown good performance in our application, where we illustrate the kinds of trends and disturbances that can be captured across these two components.[6] Furthermore, the modularity of the additive GP specification allows easy modifications to accommodate different settings. Longer spans of data may contain variation in spending along different length-scales, which may require additional SE components. There may also be several periodicities requiring additional periodic components. These can be easily included additively.

Individual-level Effects The remaining effects—recency, lifetime, and purchase number—operate at the customer-level. In most applications, we do not expect short-run shocks along these inputs. We do, however, expect monotonicity. For instance, intuitively, we expect spend probability to be generally decreasing in interpurchase time. Similarly, we expect spend probability to be generally increasing in purchase number,[7] and to be generally decreasing in customer lifetime. Furthermore, while we expect monotonicity, we also expect a decreasing marginal effect. For example, we expect a priori that the difference between having spent 5

[6]We also include simulated data examples of these effects in Web Appendix B, where we know the effects true forms, and can show that the GPPM is capable of accurately recovering them.

[7]We may not expect this in our application area, freemium video games, where there can be decreasing returns to repeat purchasing.

versus 10 days ago is quite different than the difference between having spent 95 versus 100 days ago. Together, these expected traits justify using our power mean function:

$$\alpha_{\text{R}}(r) \sim \mathcal{GP}(\lambda_{\text{R1}}(r-1)^{\lambda_{\text{R2}}}, k_{\text{SE}}(r, r'; \eta_{\text{R}}, \rho_{\text{R}})),$$
$$\alpha_{\text{L}}(\ell) \sim \mathcal{GP}(\lambda_{\text{L1}}(r-1)^{\lambda_{\text{L2}}}, k_{\text{SE}}(\ell, \ell'; \eta_{\text{L}}, \rho_{\text{L}})),$$
$$\alpha_{\text{Q}}(q) \sim \mathcal{GP}(\lambda_{\text{Q1}}(r-1)^{\lambda_{\text{Q2}}}, k_{\text{SE}}(r, r'; \eta_{\text{Q}}, \rho_{\text{Q}})).$$

Again, this specification allows for long-run monotonic behavior, even out-of-sample, as captured by the mean function, and for nonparametric deviations from this expected functional form, as captured by the SE kernel. We believe that this specification is very general and widely applicable. In some cases, however, more nuance may be required in specifying these effects to accommodate company actions that occur on these time scales. If, for instance, the company offers promotions based on loyalty, these effects will operate along the lifetime dimension. In that case, the lifetime component can be modeled similarly to the calendar time component, with an additive SE component to capture these short-run deviations from the long-run, decreasing trend embodied in the above specification. We include an example of this modification in Web Appendix B.

Heterogeneity, Random Effects, and Priors We accommodate unobserved heterogeneity by assuming that the random effect δ_i comes from a normal population distribution, i.e., $\delta_i \sim \mathcal{N}(0, \sigma^2)$. In our application, we found no significant time-invariant effects $\mathbf{z}_i$, and hence we omit $\mathbf{z}_i'\boldsymbol{\gamma}$ from our model going forward. We estimate the model in a fully Bayesian fashion, and therefore specify

priors over all unknowns, including the GP hyperparameters. We use the fact that meaningful variation in the inverse logit function occurs for inputs between -6 and 6, and hence meaningful differences in the inputs to the GPPM will also occur between -6 and 6, to select proper weakly informative Normal and Half-Normal prior distributions that give weight to variation in this range. Thus, we let the population variance $\sigma^2 \sim \text{Half-Normal}(0, 2.5)$ and the base spending rate $\mu \sim \mathcal{N}(0, 5)$. For the squared exponential hyperparameters, we specify $\eta^2 \sim \text{Half-Normal}(0, 5)$ and $\rho^2 \sim \text{Half-Normal}(T/2, T)$. For the mean function, we let $\lambda_1 \sim \mathcal{N}(0, 5)$, and let $\lambda_2 \sim \text{Half-Normal}(0, 5)$. Importantly, the fully Bayesian approach, whereby both the GP function values and their associated hyperparameters are estimated from the data, allows us to automatically infer the nature of the latent functions that drive spend propensity.

Identification We need to impose identification restrictions because of the additive structure of our model. Sums of two latent functions, such as $\alpha_1(t) + \alpha_2(t)$, are indistinguishable from $\alpha_1^*(t) + \alpha_2^*(t)$, where $\alpha_1^*(t) = \alpha_1(t) + c$, and $\alpha_2^*(t) = \alpha_2(t) - c$ for some $c \in \mathbb{R}$, as both sums imply the same purchase probabilities. To address this indeterminacy, we set the initial function value (corresponding to input $\tau = 1$) to zero for all of the latent functions, except for $\alpha_{\text{T}}^{\text{Long}}(t)$. In this sense, $\alpha_{\text{T}}^{\text{Long}}(t)$, with its constant mean function μ, captures the base spending rate for new customers, and the other components capture deviations from that, as time progresses. Whenever we implement a sum of squared exponential kernels, as in the calendar time component, we also constrain the length-scale parameters to be ordered to prevent label switching. All of these constraints are easily incorporated in our estimation algorithm, described below.

1.2.3 Estimation

We use a fully Bayesian approach for inference. For concision, let $\alpha_{ij} \equiv \alpha(t_{ij}, r_{ij}, \ell_{ij}, q_{ij})$, which in our specification, is equivalent to $\alpha_{ij} = \alpha_{\mathrm{T}}^{\mathrm{Long}}(t_{ij}) + \alpha_{\mathrm{T}}^{\mathrm{Short}}(t_{ij}) + \alpha_{\mathrm{T}}^{\mathrm{DoW}}(t_{ij}) + \alpha_{\mathrm{R}}(r_{ij}) + \alpha_{\mathrm{L}}(\ell_{ij}) + \alpha_{\mathrm{Q}}(q_{ij})$. To further simplify notation, we let the independent components of the sum be indexed by k, with generic inputs τ_k, such that this GP sum can be written as $\alpha_{ij} = \sum_{k=1}^{K} \alpha_k(\tau_{k_{ij}})$. Each of these components is governed by a set of hyperparameters, as outlined in the previous section, denoted here as $\boldsymbol{\phi}_k$, with the collection of all hyperparameters denoted $\boldsymbol{\phi}$. Finally, for each component, we let the vector of function values over all possible inputs along that dimension be denoted as $\boldsymbol{\alpha}_k$. With this simplified notation, the joint density of the data and the model unknowns is:

$$p(\boldsymbol{y}, \{\boldsymbol{\alpha}_k\}, \boldsymbol{\delta}, \boldsymbol{\phi}, \sigma^2) = \left[\prod_{i=1}^{I}\prod_{j=1}^{M_i} p(y_{ij}|\alpha_{ij}, \delta_i)p(\delta_i|\sigma^2)\right]\left[\prod_{k=1}^{K} p(\boldsymbol{\alpha}_k|\boldsymbol{\phi}_k)\right]p(\sigma^2)p(\boldsymbol{\phi}). \tag{1.10}$$

As the full posterior distribution $p(\{\boldsymbol{\alpha}_k\}, \boldsymbol{\delta}, \boldsymbol{\phi}, \sigma^2|\boldsymbol{y})$ is not available analytically, we use Markov Chain Monte Carlo Methods (MCMC) to draw samples of the unknown function values, random effects, population parameters, and GP hyperparameters from the posterior.

Hamiltonian Monte Carlo As the function values and the hyperparameters do not have closed-form full conditionals, our setup is non-conjugate, and Gibbs sampling is not an option. Moreover, as the function values and the hyperparameters typically exhibit strong posterior dependence, ordinary Metropolis-Hastings procedures that explore the posterior via a random walk are not efficient. We therefore use the Hamiltonian Monte Carlo (HMC) algorithm that leverages the gradient of the posterior to direct the exploration of the Markov chain to avoid

random-walk behavior. HMC methods are ideal for non-conjugate GP settings such as ours, as they can efficiently sample both the latent function values as well as the hyperparameters (Neal, 1998). In particular, we use the No U-Turn Sampling (NUTS) variant of HMC to draw samples of the unknown function values $\boldsymbol{\alpha}_k$, customer-specific random effects $\boldsymbol{\delta}$, population parameters σ^2, and the GP hyperparameters $\boldsymbol{\phi}$, from the posterior. For completeness, we include a brief overview of HMC here, and refer the reader to Neal (2011) for further details.

HMC is a variant of the Metropolis-Hastings algorithm that uses a proposal distribution that is based on the Hamiltonian dynamics of a particle moving in a potential field. Suppose our interest is in sampling a set of parameters $\boldsymbol{\theta} \in R^p$ (i.e., particle positions) from a target posterior distribution $p(\boldsymbol{\theta}|\boldsymbol{y})$. For our model, $\boldsymbol{\theta}$ can contain the entire set of unknown function values and GP hyperparameters. HMC uses a vector of auxilliary momentum variables $\boldsymbol{\zeta} \in R^p$ drawn from a multivariate normal $N(\boldsymbol{\zeta}|0, M)$ where the covariance matrix M is the mass matrix. Both the positions and the momentum variables are jointly sampled from a joint density $p(\boldsymbol{\theta}, \boldsymbol{\zeta}|\boldsymbol{y}) = p(\boldsymbol{\theta}|\boldsymbol{y})p(\boldsymbol{\zeta})$. The values of $\boldsymbol{\theta}$ are retained, where as the samples of $\boldsymbol{\zeta}$ are ignored. Algorithm 1 outlines a single HMC iteration.

As can be seen from Algorithm 1, each iteration of the HMC algorithm involves several leapfrog steps in which $\boldsymbol{\theta}$ and $\boldsymbol{\zeta}$ evolve according to a discretization of Hamilton's equations. The HMC sampler uses the gradient of the log-posterior to direct the exploration of the posterior. This allows it to avoid the random walk behavior of ordinary Metropolis-Hastings procedures and it therefore traverses the posterior in an efficient fashion. HMC methods are ideal for non-conjugate GP settings such as ours, as they can efficiently sample both the latent function values as well as the hyperparameters.

In practice, we need to specify values for the step size ϵ, the number of

Algorithm 1 HMC Iteration (Given stepsize ϵ, number of leapfrog steps, L mass matrix M, and $\boldsymbol{\theta}_{current}$)

Initialize $\boldsymbol{\theta}_{(0)} \leftarrow \boldsymbol{\theta}_{current}$, $\boldsymbol{\zeta}_{(0)} \sim \mathcal{N}(0, M)$

for $l = 0, \ldots, L-1$ **do** ▷ Perform Leapfrog steps
  $\boldsymbol{\zeta}_{(l+1/2)} \leftarrow \boldsymbol{\zeta}_{(l)} + \frac{1}{2}\epsilon \nabla_\theta \log p(\boldsymbol{\theta}_{(l)}|\boldsymbol{y})$
  $\boldsymbol{\theta}_{(l+1)} \leftarrow \boldsymbol{\theta}_{(l)} + \epsilon M^{-1} \boldsymbol{\zeta}_{(l+1/2)}$
  $\boldsymbol{\zeta}_{(l+1)} \leftarrow \boldsymbol{\zeta}_{(l+1/2)} + \frac{1}{2}\epsilon \nabla_\theta \log p(\boldsymbol{\theta}_{(l+1)}|\boldsymbol{y})$
end for

$r = \min\left[1, \frac{p(\boldsymbol{\theta}_{(L)}|\boldsymbol{y})\, p(\boldsymbol{\zeta}_{(L)})}{p(\boldsymbol{\theta}_{(0)}|\boldsymbol{y}) p(\boldsymbol{\zeta}_{(0)})}\right]$ ▷ Compute acceptance probability
$u \sim \text{Uniform}(0, 1)$ ▷ Uniform draw
if u $< r$, **then** return $\boldsymbol{\theta}_{(L)}$ ▷ Accept or reject proposal
else return $\boldsymbol{\theta}_{(0)}$
end if

leapfrog steps L and the mass matrix M, and finding the right set of values for these can be sometimes challenging. We therefore use the No U-Turn Sampling (NUTS) variant of HMC as implemented in the Stan probabilistic programming language (Hoffman and Gelman, 2014a; Carpenter et al., 2017). Stan uses an adaptive version of the HMC algorithm wherein ϵ, L and M are updated across the MCMC iterations to ensure rapid mixing, while still maintaining detailed balance. Since each iteration of HMC involves multiple leapfrog steps, an HMC iteration is not directly comparable to that of the ordinary Metropolis-Hastings algorithm, and convergence is achieved in much fewer MCMC iterations. Details of NUTS are given in Hoffman et al. (2014).

Implementation We implement NUTS using the Stan probabilistic programming language (Hoffman and Gelman, 2014a; Carpenter et al., 2016). Stan has recently gained traction as an efficient and easy-to-use probabilistic programming tool for Bayesian modeling. We use Stan as it is an efficient implementation of adaptive HMC. Stan programs are simple to write and modify, and therefore facilitate easy

experimentation, without the need for extensive reprogramming. This is important for the wider adoption of this framework in practice.[8] Finally, given the efficiency of HMC and Stan, convergence, as measured by the $\hat{R}$ statistic (Gelman and Rubin, 1992), is achieved in as few as 400 iterations, although in this paper all estimation is done with 4,000 iterations with the first 2,000 used for burn-in.

1.3 Application

We apply our framework to understand the spending dynamics in two free-to-play mobile games from one of the world's largest video game companies. The data take the form of simple spend incidence logs, with user IDs and time stamps.[9] In free-to-play (or "freemium") settings, users can install and play video games on their mobile devices for free, and are offered opportunities to purchase within the game. These spend opportunities typically involve purchasing in-game currency, like coins, that may subsequently be used to progress more quickly through a game, obtain rare or limited edition items to use with their in-game characters, or to otherwise gain a competitive edge over non-paying players. Clearly, the nature of these purchases will depend on the game, which is why it is important for a model of spending behavior to be fully flexible in its specification of the regular, underlying drivers of purchasing. We cannot name the games here because of non-disclosure agreements. Instead, we use the general descriptors Life Simulator (LS) and City Builder (CB) to describe the games.

The games and ranges of data used were selected by our data provider, in an

[8] Our Stan code is available online.

[9] There is no personally identifiable information in our data; player information is masked such that none of the data we use or the results we report can be traced back to the actual individuals. We also mask the identification of the company as per their request.

effort to understand spend dynamics over specific periods of time. We use a random sample of 10,000 users for each of the two games. Each sample is drawn from users who installed the game within the first 30 days, and spent at least once during the training window. We used 8,000 users for estimation, and 2,000 for cross validation. In the Life Simulator (LS) game, players create an avatar, then live a digital life as that avatar. Purchases in this context can be rare or limited edition items to decorate or improve their avatar or its surroundings. Often times, limited edition items are themed according to holidays such as Christmas or Halloween. Our data come from a 100 day span of time covering the 2014 Christmas and New Year season. In the City Builder (CB) game, players can create (or destroy) a city as they see fit. Customers make purchases to either speed up the building process or to build unique or limited edition additions to their cities. Our data come from an 80 day period of time at the start of 2015, at the tail end of the Christmas and New Year holidays.

The time series of spending for the two games are shown in Figure 1.2. We have also marked specific time periods of interest to the company, which we will discuss in more detail in our analysis. From these figures, it is difficult to parse out what exactly is driving the aggregate pattern of purchases. The figure includes customers who installed the game any time within the first 30 day window. Typically, customers are most active when they start playing a game, so we expect to see more spending in the first 30-40 days simply because there are likely more people playing in that period, and new players are entering the pool of possible spenders. This rise and subsequent fall is, in essence, the joint impact of the recency, lifetime, and purchase number effects. We see, however, that even the general rise-fall pattern varies across the two games. This could be due to different patterns in these underlying drivers of spending, or it could be due to the influence of calendar time events. In essence, it is unclear what else underlies the aggregate

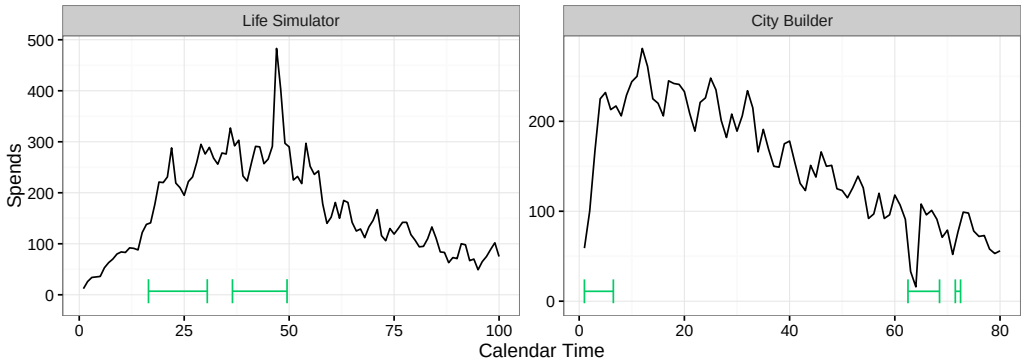

Figure 1.2: Spend incidence by day (calendar time) in each game. Bars indicate time periods of interest, as specified by the company, and as discussed more in Section 3.2.1.

spends.

We also see many peaks and valleys in spending over the entire time horizon, the significance of which cannot be diagnosed without deeper analysis. For example, it is difficult to discern which "bumps" in the plots are meaningful, and which represent random noise. If 5,000 players are active at any given day, then a jump of 50 spends in may represent a random fluctuation. In contrast, if only 1,000 players are active, the same jump of 50 spends may be very meaningful. In other words, the significance of a particular increase in spending depends on how many customers are still actively spending at that time, which in turn depends on the individual-level recency, lifetime, and purchase number effects. An accurate accounting of the impact of calendar-time events cannot be made without considering these individual-level predictors of spending, and it is thus important to develop a model-based understanding of the underlying spend dynamics, which is what we do via the GPPM.

1.3.1 Model Output and Fit

The GPPM offers a visual and highly general system for customer base analysis that is driven by nonparametric latent spend propensity functions. These latent curves are the primary parameters of the model, and their posterior estimates are displayed in Figure 1.3 for LS, and in Figure 1.4 for CB. We call these figures the GPPM dashboards, as they visually represent latent spend dynamics. As we will see in 1.3.2, these dashboards can be used to accomplish many of the goals we have discussed throughout the previous sections, including forecasting spending, understanding purchasing at the individual-level, assessing the influence of calendar time events, and comparing spending patterns across products.

These dashboards are underpinned by a set of hyperparameters, and estimated jointly with a random effects distribution capturing unobserved heterogeneity. Posterior medians of these parameters are displayed in Table 1.1. While the hyperparameters summarize the traits of the estimated dashboard curves, as explained in Section 2.1, we can gain a greater understanding of the dynamics from an analysis of the estimated dashboard curves themselves, as we do in the subsequent sections. The other parameters in Table 1.1 are the base spending rate, μ, and the population variance of the random effects distribution, σ^2, which reflects the level of heterogeneity in base spend rates estimated in each customer base.

Model Fit First, to validate our model, we look at its fit to the observed daily spending data, both in the calibration sample of 8,000 customers and in the holdout sample of 2,000 customers. A closed-form expression is not available for the expected number of aggregate counts in the GPPM. We therefore simulate spending from the posterior predictive distribution by using the post convergence HMC draws

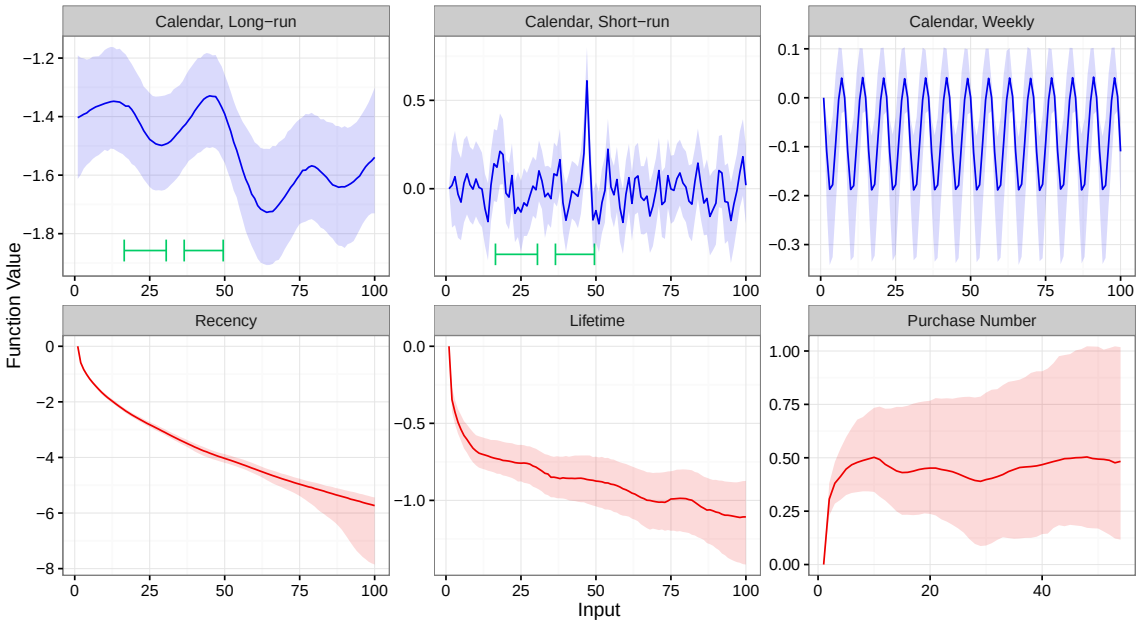

Figure 1.3: Posterior dashboard for the Life Simulator customer base. Curves are the median posterior estimates for the latent components of $\alpha(t, r, \ell, q)$ with 95% credible intervals. The blue plots (top row) are the calendar time components, while the red (bottom row) are the individual-level effects. The marked time periods (green bars) are areas of interest to the company, as discussed in Section 3.2.1.

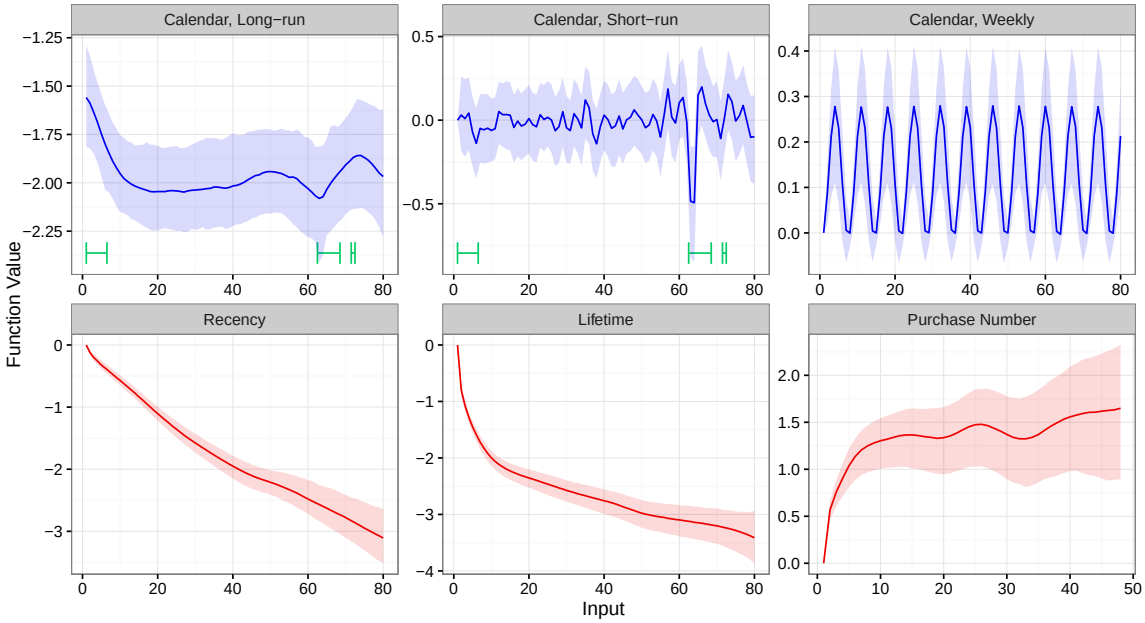

Figure 1.4: Posterior dashboard for the City Builder customer base. Curves are the median posterior estimates for the latent components of $\alpha(t, r, \ell, q)$ with 95% credible intervals. The blue plots (top row) are the calendar time components, while the red (bottom row) are the individual-level effects. The marked time periods (green bars) are areas of interested to the company, as discussed in Section 3.2.1.

Component		LS	CB	Component		LS	CB
Cal, Long	η_{TL}	0.17	0.22	Lifetime	η_{L}	0.06	0.23
	ρ_{TL}	11.75	10.32		ρ_{L}	9.77	12.25
Cal, Short	η_{TS}	0.15	0.16		λ_{L1}	-0.34	-0.75
	ρ_{TS}	1.11	1.29		λ_{L2}	0.25	0.36
Cal, DoW	η_{TW}	1.08	1.19	Purchase Number	η_{Q}	0.10	0.20
	ρ_{Q}	9.17	9.59		ρ_{Q}	4.93	5.36
Recency	η_{R}	0.04	0.10		λ_{Q1}	0.28	0.52
	ρ_{R}	10.23	11.05		λ_{Q2}	0.15	0.30
	λ_{R1}	-0.59	-0.13	Base Rate	μ	-1.49	-1.92
	λ_{R2}	0.49	0.72	Heterogeneity	σ^2	0.68	0.93

Table 1.1: Posterior median parameter estimates for both games.

for each parameter, including the latent curves and random effects. The top row of Figure 1.5 shows the actual spending and the median simulated purchase counts (dashed line) for the two games, along with 95% posterior predictive intervals.

We see that the fit is exceptional, and tracks the actual purchases almost perfectly in both cases. This is not surprising, as we model short-run deviations in the probability of spending on a daily basis and therefore essentially capture the residuals from the smoother model components. That is, the short-run calendar time component captures any probability that is "left-over" from the other components of the model, enabling us to fit in-sample data exceptionally well. To test that the model does not overfit the in-sample day-to-day variability, we explore the simulated fit in the validation sample of 2,000 held-out customers. The bottom row of Figure 1.5 shows that the fit to this sample is still excellent, although not as perfect as in the top row. While the probabilistic residuals from the calibration data are not relevant for the new sample, much of the signal present in the calendar time trends and the individual-level effects continue to matter, thus contributing to the good fit.

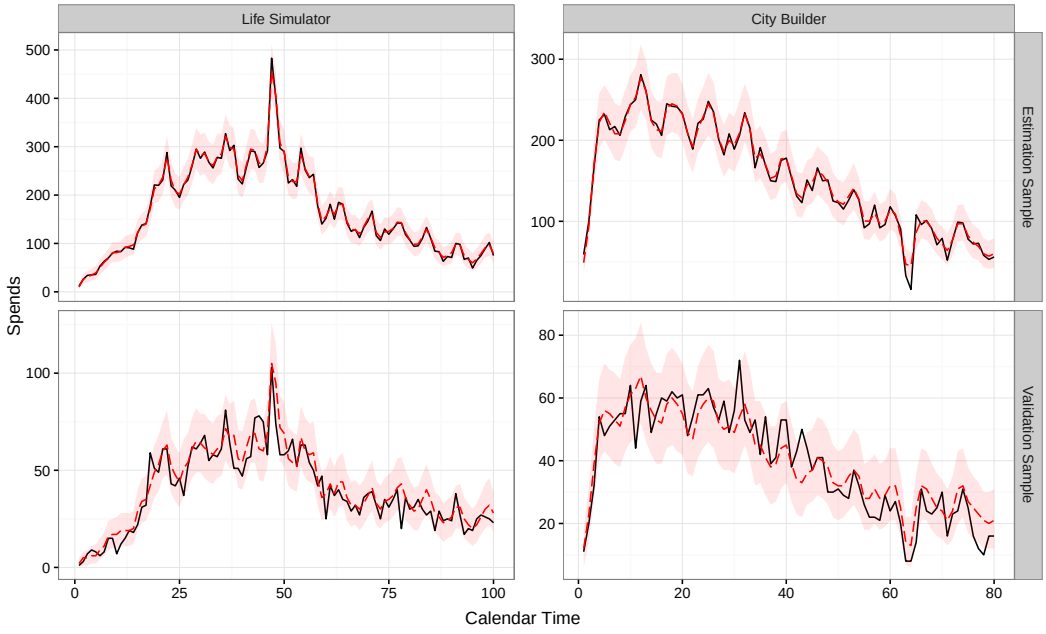

Figure 1.5: True and simulated spending by day under the GPPM with 95% posterior predictive intervals. The black is the data while the red (dashed) is the median simulated fit. In the top row, we show the fit in the estimation data of 8,000 customers, where the two curves are nearly indistinguishable, while in the bottom row, we show the fit in the validation sample of 2,000 held-out customers.

Fit Decomposition To better understand how the latent curves in the dashboard contribute to the fits seen in Figure 1.5, we now break down that fit along our latent dimensions. For that, we focus on the LS game. Our main focus is on assessing how much of the day-to-day spending is explained by the calendar time components of the model versus the typically smoother, individual-level recency, lifetime, and purchase number components. To do that, we examine how the fit changes when different components of the model are muted. We "mute" a component by replacing it with a scalar that is equal to the average of its function values over all its inputs. Note that we do not re-estimate a model when we mute a component; instead, muting allows us to see how much of the overall fit is driven by a given component.

The fit decomposition is shown in Figure 1.6. Overlaid on the true spending time series, we have three muted fits: in the first, we mute the short-run calendar time component; in the second, we mute both the short and long-run calendar time

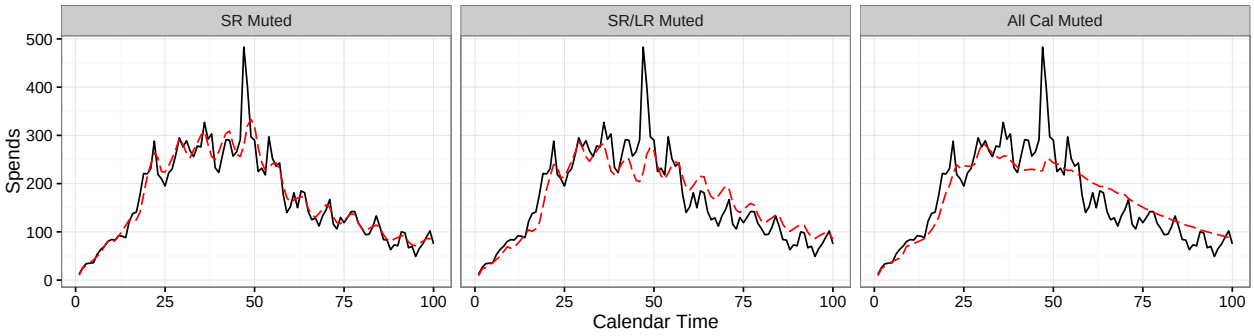

Figure 1.6: Fit decomposition on the LS spending data. Each panel from left to right represents muting an additional component of the model; the worsening fit shows how much of the full model fit is driven by the muted component.

components; and in the third, we mute all calendar time components. From the continued good fit of the muted models, we can see that the majority of the full model fit is actually driven by the individual-level predictors of spend: recency, lifetime, and purchase number. This finding is largely in keeping with the established literature on customer base analysis, which has robustly shown that models based on these components can do well at fitting and forecasting spend activity. However, we also find that calendar time plays a non-negligible role: while the short-run component generally captures the residuals, as explained before, the long-run component plays an important role in capturing changes in base spending rates over time. Furthermore, the cyclic component, which is a highly predictable yet novel element of our model, plays an important role in explaining day-to-day variability in spending.

1.3.2 Dashboard Insights

While fit validates the utility of the GPPM, one of the primary motivations of the model is to provide managers with a model-based decision support system that captures effects of interest, and allows for a visual understanding of the drivers of spend behavior. Thus, the key output of our model is the GPPM dashboard

(Figures 1.3 and 1.4), which portrays the posterior estimates of the latent propensity functions. These latent spend propensity curves are readily interpretable, even by managers with minimal statistical training. We illustrate here the insights that managers can obtain from these model-based Dashboards.

Calendar Time Effects

Events that happen in calendar time are often of great importance for managers, but their impact is often omitted from customer base analysis models. The GPPM includes these effects nonparametrically through the calendar time components of the model, such that impact of calendar time events is captured flexibly and automatically. Calendar time effects are estimated jointly with the individual-level drivers of spending, recency, lifetime, and purchase number. This means the impact of calendar time on propensity to spend is assessed only after controlling for these drivers of respend behavior, which account for the natural ebb and flow of spending, including dynamics in the numbers of active customers.

Importantly, capturing the impact of calendar time events requires no inputs from the marketing analyst, as would be required in a model where time-varying covariates are explicitly specified. This implies that their presence and significance must be evaluated ex post facto. This has many benefits: first, even in the face of information asymmetries or unpredictable shocks, the events will be captured by the GPPM. Second, the shape of the impact of these events is automatically inferred, rather than assumed. Finally, because the impact is captured by changes in the calendar time components of the propensity model, their impact can be assessed visually. We demonstrate the analysis of calendar time events using our two focal games. The top row of plots in each dashboard (colored blue) represents the calendar time effects. From left to right, we have the long-run trends, short-run

shocks, and periodic day of the week effects. Beneath these curves, we have placed bars indicating time periods of interest to the company.

Life Simulator Events Two events of note occurred in the span of the data. The first marked time period $t \in [17, 30]$ corresponds to a period in which the company made a game update, introduced a new game theme involving a color change, and also donated all proceeds from the purchases to a charitable organization. The second marked period, around $t \in [37, 49]$, corresponds to another game update that added a Christmas-themed quest to the game, with Christmas itself falling at $t = 48$, right before the end of the holiday quest.

From the dashboard, we learn several things: first, there is a prominent spike in short-run spending the day *before* Christmas. This Christmas Eve effect illustrates that events do not have to be anticipated to be detected in the model, and we illustrate in the subsequent section how the GPPM parses out the impact of short-run events, using this effect as the example. In the long-run curve, we see a decrease in spending coinciding with the charity update, an increase in spending coinciding with the holiday event, and then a significant drop-off subsequent to the holiday season. Without a longer range of data, it is hard to assess the meaning of these trends. It does appear that the charity event lowered spend rates. The impact of the holidays is more unclear: it could be that the holiday game update elevated spending, and then as time went on, spend levels returned to normal. Alternatively, spend levels could be elevated simply due to the holiday season, with a post-holiday slump that is unrelated to the game updates. Although we cannot conclusively parse out these stories, we can tell that calendar time dynamics are at play, and appear linked to both real world shocks and company actions.

City Builder Events The marked areas of the CB dashboard in Figure 1.4 again correspond to events of interest. The start of the data window ($t \in [1, 6]$) coincides with the tail end of the holiday season, from December 30 to January 4. Another event begins at $t = 63$, when the company launched a permanent update to the game to encourage repeat spending. We mark five additional days after that update to signify a time period over which significant post-update activity may occur. Finally, at $t = 72$, there was a crash in the app store.

We see, as in the previous game, that the spending level in the holidays ($t \in [1, 6]$) was quite high and fell dramatically subsequently. This lends some credence to a general story of elevated holiday season spending, as there was no game update in CB during this time. Spending over the rest of the time period was relatively stable. The update that was intended to promote repeat spending had an interesting effect: there was an initial drop in spending, most likely caused by reduced playtime on that day because of the need for players to update their game or because of an error in the initial launch of the update. After the update, an uptick in long-run spending is observable, but this was relatively short-lived. Finally, we find no effect for the supposed app store crash, which in theory should have prevented players from purchasing for the duration of the crash. It is plausible that the crash was for a short duration or occurred at a time when players were not playing.

Day of the Week Effects Across both games, we note the significance of the periodic day of the week effect. In both cases, spend propensity varies by day of the week by a magnitude of 0.3. For comparison, the long-run calendar time effect of LS has a range of 0.5, while that of CB has a range of 0.6. The magnitude of the periodic effect serves to re-emphasize a point already made in the fit decomposition:

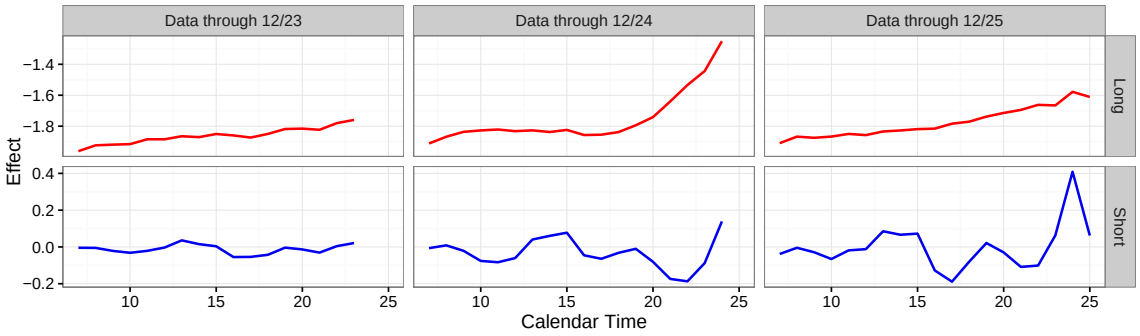

Figure 1.7: Event detection in the GPPM. From left to right, we add daily data, and see how the impact of Christmas Eve is separated between the long-run (top, red) and short-run (bottom, blue) calendar time curves.

a large amount of the calendar time variability in spending can be attributed to simple predictable cyclic effects, something customer base models have previously ignored, but that can be powerful in forecasting future purchase behavior.

Event Detection Often, calendar time events are unknown a priori, but can significantly affect consumers' spending rates in the short-run. The short-run function is capable of automatically detecting and isolating these disturbances. That is, if something disrupts spending for a day, such as a crash in the payment processing system, or an in-game event, it will be reflected either as a trough or as a spike in the short-run function, as evident for example in the Christmas Eve effect in LS. In this section, we illustrate how this works in practice.

The GPPM estimation process decomposes the calendar time effect along sub-functions with differing length-scales. As such, when there is a disturbance, the GPPM must learn the relevant time scale for the deviation—here, either short or long-term—and then adjust accordingly. We illustrate this dynamically unfolding adjustment process for the LS Christmas Eve effect in Figure 1.7 by estimating the model using progressively more data over the range 12/23/2014 to 12/25/2014. The different columns of the figure show how the long-run (top row) and the short-run

(bottom row) components vary when data from each successive day is integrated into the analysis. The second column shows the impact of adding the data from Christmas Eve. An uptick in spending is apparent, but the GPPM cannot yet detect whether this uptick will last longer or just fade away. The day after (third column), it becomes clear from looking at the long-run and short-run plots that the effect was only transient, which is reflected clearly in the short-run curve.

This example illustrates that the GPPM can capture effects of interest with no input from the analyst, and that the nature of this effect is visually apparent in the model-based dashboard within days of its occurrence. Note that, importantly, each column of Figure 1.7 represents a re-estimation of the GPPM, using the past day's data; event detection can only occur at the level of aggregation of the data (in this case, daily), upon re-estimation of the model. Nonetheless, this capability can be immensely valuable to managers in multiproduct firms where information asymmetries abound. For example, in digital contexts, product changes can sometimes be rolled out without the knowledge of the marketing team. Similarly, disruptions in the distribution chain can occur with little information filtering back to marketing managers. The GPPM can capture the impact of such events automatically and quickly, isolate them from the more regular, predictable drivers of spending, and bring them to the attention of managers.

Individual-level Effects

While the inclusion of calendar time effects is a key innovation in our model, the primary drivers of respend behavior are the individual-level recency, lifetime, and purchase number effects. We can see this both through the fit decomposition, where much of the variability in spending is captured even when the calendar time effects are muted, and also by assessing the range of the effects in the dashboard. As

mentioned in Section 1.2.2, the range of relevant inputs in an inverse logit framework is from -6 to 6. For propensity values $\alpha < -6$, the respend probability given by $\text{Logit}^{-1}(\alpha)$ is approximately 0. Similarly, for propensity values $\alpha > 6$, the respend probability is approximately 1. This gives an interpretability to the curves in the dashboard, as their sum determines this propensity, and hence their range determines how much a given component of the model can alter expected respend probability. Relative to the calendar time effects, we can see in the dashboard that the ranges of the individual-level effects are significantly larger, implying that they explain much more of the dynamics in spend propensity than the calendar time components.

Recency and Lifetime In both of our applications, the recency and lifetime effects are smooth and decreasing as expected. For managers, this simply means that the longer someone goes without spending, and the longer someone has been a customer in these games, the less likely that person is to spend. The recency effect is consistent with earlier findings and intuitively indicates that if a customer has not spent in a while, he or she is probably no longer a customer. The lifetime effect is also expected, especially in the present context, as customers are more likely to branch out to other games, with the passage of time. More interesting are the rates at which these decays occur, and how they vary across the games. These processes appear to be fundamentally different in the two games. In LS, the recency effect has a large impact, whereas the lifetime effect assumes a minimal role. In contrast, in CB, both appear equally important. These results may be a result of, for example, the design of the product (game), which encourages a certain pattern of purchasing.

Purchase Number The purchase number effect also appears different across the games. In LS, the effect seems relatively insignificant: although there is initially a

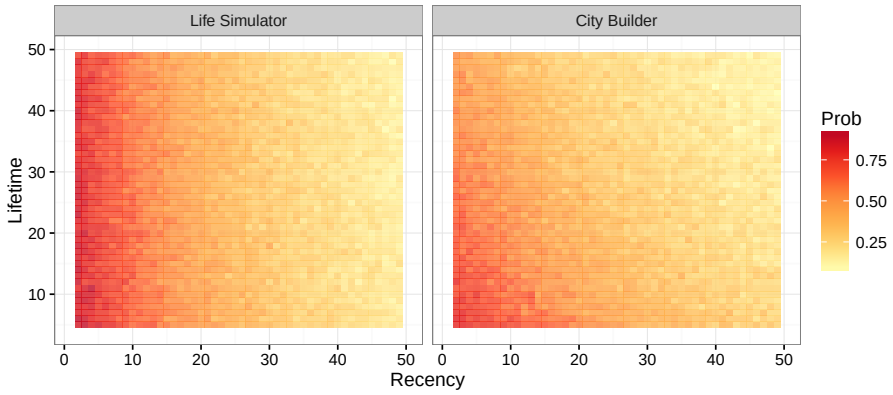

Figure 1.8: Respend probability heat maps for a customer with $q = 3$ and $\delta_i = 1$. Colors represent the probability of respending in the next 100 days, given the current recency and lifetime values. Note that some pairs of recency and lifetime that are displayed in the plot are not realistic: a customer cannot have recency higher than lifetime.

slight rise, it quickly evens out, with a large confidence interval. In CB, the effect appears quite significant: it is generally increasing, but again appears to flatten out toward the end. The effect in CB is more consistent with our expectations: significant past purchasing should indicate a loyal customer, and a likely purchaser. A mild or neutral effect, like seen in LS, may indicate decreasing returns to spending in the game, or a limited number of new items that are available for purchase, such that the customer quickly runs out of worthwhile purchase opportunities.

Behavioral Implications The shapes of these curves have implications for player behavior and for designing general CRM strategies. In LS, the recency effect is the primary predictor of churn: if a customer has not spent for a while, she is likely no longer a customer. On the other hand, the lifetime effect seems to operate only in the first few days of being a customer, then levels out. This implies that customers are most likely to spend when they are new to the game, within roughly two weeks of their first purchase. In contrast, in CB, the effects are more equal in magnitude, and more gradual. The customers that are least likely to spend again are those that have been customers the longest, and have gone the longest without spending.

We illustrate these differences here via an individual-level analysis of respend probability. Specifically, we ask the question, given an individual's recency and lifetime, what is the probability that she spends again in the next 100 days? To carry out this simulation, we fix the calendar time effect to its average value, and assume that the individual has already spent three times. The results of the simulation are displayed in Figure 1.8, and re-emphasize the point that recency explains much of the respend probability in LS, while lifetime and recency are both relevant in CB. This analysis also emphasizes the idea that, while the dynamic effects in the GPPM are the same for all customers, different positions in the individual-level subspace $(r_{ij}, \ell_{ij}, p_{ij})$ are associated with very different expected future purchasing behavior.

In summary, we have seen that the GPPM weaves together the different model components in a discrete hazard framework, and offers a principled approach for explaining aggregate purchase patterns based on individual-level data. The model-based dashboard generated by the GPPM is not the result of ad hoc data smoothing, but arises from the structural decomposition of spend propensity via the different model components. The GPPM jointly accounts for both the predictable individual-level determinants of respend probability, such as recency, lifetime, and purchase number, and calendar time events along multiple length-scales of variation. It is therefore able to flexibly represent the nature of customer respend probability, as well as accurately portray the existence and importance of calendar time events and trends.

1.3.3 Predictive Ability and Model Comparison

Apart from interest in understanding past spending dynamics, managers also need to forecast *future* purchasing activity. Although the primary strength of the GPPM is in uncovering latent dynamics, and conveying them in an intuitive fashion through the model-based dashboard, the GPPM also does very well in predicting future spending. Just as in-sample fit was driven by the recency, lifetime, and purchase number components, predictive performance depends primarily on the ability to forecast these components for observations in the holdout data. While forms of recency, lifetime, and purchase number effects are incorporated in most customer base models, the isolation of these effects apart from transient calendar time variability, along with nonparametric characterization of these predictable components, and the inclusion of the cyclic component, allow the GPPM to significantly outperform benchmark customer base analysis models in predictive ability.

In this section, we focus on comparing both model fit and future predictive performance, and therefore reestimate the GPPM by truncating our original calibration data of 8,000 customers along the calendar time dimension. In particular, we set aside the last 30 days of calendar time activity to test predictive validity. Forecasting with the GPPM involves forecasting the latent functions that comprise it. In forecasting these latent functions, we use the predictive mechanisms outlined in Section 2.1 (Equation 1.4). As the holdout data is constructed by splitting the original dataset along the calendar time dimension, a substantial number of the observations in the holdout data contain recency, lifetime, and purchase number values that are within the observable range of these variables in the calibration dataset. This is especially true for observations belonging to newly

acquired customers. However, for the oldest customers, the individual-level curves need to be forecast.

Benchmark Models

We compare predictive performance of the GPPM with that of a number of benchmark models. Many individual-level models have been developed to do customer base analysis. At its core, the GPPM is a very general discrete hazard model and as such it can be compared to other hazard models for interpurchase times (Gupta, 1991; Seetharaman and Chintagunta, 2003). Similarly, given its reliance on recency, lifetime, and purchase number dimensions of spending, the GPPM is closely related to traditional customer base analysis models for non-contractual settings of the "buy-till-you-die" (BTYD) vein (Schmittlein et al., 1987; Fader et al., 2005, 2010). Finally, the discrete hazard approach could be modified with a different specification of the spend propensity.

Hazard Models We consider two standard discretized hazard models: the *Log-Logistic* model and the *Log-Logistic Cov* model, which are standard log-logistic hazard models without and with time-varying covariates respectively. We choose the log-logistic hazard as it can flexibly represent both monotonic and non-monotonic hazard functions. In the model with covariates, we use indicator variables over the time time periods of interest indicated at the start of Section 3. In estimating both of these models, we employ the same Bayesian estimation strategy, using Stan, with the same random effect heterogeneity specification as in the GPPM.

BTYD We use the *Pareto-NBD* (Schmittlein et al., 1987) and the *BGNBD* (Fader et al., 2010) as benchmarks in this class. While many variants of BTYD have been

developed over the years, the Pareto-NBD has stood the test of time as the gold standard in forecasting power in non-contractual settings, often beating even more recent models (see, e.g., the PDO model in Jerath et al. (2011)). The BGNBD is a more discrete analogue of the Pareto-NBD, where customer death can occur after each purchase, rather than continuously.[10]

Propensity Models In this case, we retain the discrete time hazard inverse logit framework, while altering the specification of the dynamics. In particular, we explore two specifications: the *Linear Propensity Model (LPM)* and the *State Space Propensity Model (SSPM)*. These models have not been explored elsewhere in the literature; we include them here to help understand the benefits of the GP approach to modeling dynamics.

In the LPM, we remove the nonparametric specification altogether, and instead model all effects linearly, as
$\Pr(y_{ij} = 1) = \text{logit}^{-1}(\mu + \beta_1 t_{ij} + \beta_2 r_{ij} + \beta_3 \ell_{ij} + \beta_4 q_{ij} + \delta_i)$. This is the simplest discrete hazard model specification that includes all of our time scales and effects.

In the SSPM, we explore an alternate nonparametric specification for the dynamic effects. There are a number of competing nonparametric function estimation techniques, including dynamic linear models and various spline specifications, and there are technical links between many of these modeling approaches. Moreover, within each of class of models, there is a range of specifications that are possible, making the choice of a suitable benchmark difficult. We chose to implement a state space specification that is roughly equivalent to the GP structure in our main model. Specifically, we again decompose the propensity function $\alpha(t, r, \ell, q)$ into additive components along each dimension. For the

[10]We estimate these models using the BTYD package in the R programming language.

calendar time dimension, just as in the GPPM, we make no assumptions about its behavior, and hence model it as a random walk:

$$\alpha_{\mathrm{T}}(t) = \alpha_{\mathrm{T}}(t-1) + \epsilon_{\mathrm{T}t}, \;\; \epsilon_{\mathrm{T}t} \sim \mathcal{N}(0, \zeta_{\mathrm{T}}^2). \tag{1.11}$$

For the other dimensions, we assume as in the GPPM that there will likely be monotonicity, and hence include a trend component. This leads to a local level and trend specification:

$$\alpha_d(\tau) = \alpha_d(\tau-1) + \gamma_d(\tau) + \epsilon_{d\tau}, \;\; \epsilon_{d\tau} \sim \mathcal{N}(0, \zeta_d^2), \tag{1.12}$$

$$\gamma_d(\tau) = \gamma_d(\tau-1) + \xi_{d\tau}, \;\; \xi_{d\tau} \sim \mathcal{N}(0, \psi_d^2). \tag{1.13}$$

Interestingly, when used with a Gaussian observation model (meaning the data generating process is $\mathcal{N}(\alpha(\tau), \nu^2)$ instead of our latent propensity formulation), the local level and trend model has links to cubic spline smoothing (Durbin and Koopman, 2012). In addition to the above specified components, we also included a cyclic function of calendar time to mirror the GP periodic kernel component, as well as the random effects.

Forecasting Results

The re-estimated in-sample fit and the out-of-sample forecast of the GPPM for both games are displayed in Figure 1.9. Again, the dashed lines represent medians, while the intervals represent 95% posterior predictive intervals. We see that, again, the GPPM fits very well in-sample, but importantly also fits well in the holdout period. Out-of-sample, we see smooth decreasing trends in both games, together with the predictable day of the week effect. Referring back to Figure 1.6, we see that the forecast fit is very similar to the fit decomposition with no short and long-run

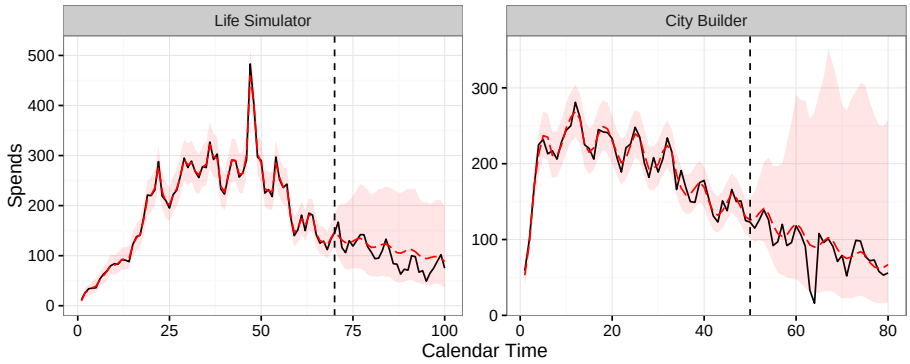

Figure 1.9: GPPM daily spending forecast. The data is in black with the median simulated GPPM fit in red (dashed) and 95% posterior predictive intervals. The holdout period is the last 30 days of data, demarcated by the dashed line.

components. This is because, far from the range of the data, components modeled with a stationary kernel will revert to their mean function, which for the calendar time effects is constant, effectively muting them far into the holdout period. How long it takes for this reversion to happen depends on the smoothness of the estimated function.

Table 1.2 shows the predictive performance of the GPPM and all of our benchmark models. The table reports the mean absolute percentage error (MAPE) and the root mean squared error (RMSE) for the calibration and holdout datasets. Several of our benchmark fits are displayed in Figure 1.10. Crucially, the fit of the GPPM is almost always significantly better than the benchmarks, both in and out-of-sample. We proceed to briefly analyze each of the benchmarks, and give intuition for why the GPPM outperforms them.

The log-logistic hazard models perform particularly poorly. In fact, the fit of the log-logistic models using the full range of the data is worse than forecast fit of the GPPM; thus, we did not re-estimate the log-logistic models in a separate forecasting task. Neither of these models captures the lifetime and purchase number drivers of spending, which are typically highly predictive of spending. Furthermore,

	Life Simulator			City Builder		
	Overall	In-sample	Holdout	Overall	In-sample	Holdout
GPPM	0.09	0.03	0.24	0.15	0.05	0.32
	13.25	5.74	22.54	15.00	9.79	20.97
Log-Logistic	0.42	0.31	0.67	0.41	0.19	0.77
	68.27	71.75	59.35	46.78	46.91	46.55
LL Covs	0.28	0.19	0.48	0.27	0.15	0.48
	62.81	67.22	51.04	36.28	32.78	41.47
Pareto-NBD	0.24	0.20	0.33	0.27	0.16	0.45
	45.10	49.64	32.10	33.54	36.56	27.80
BGNBD	0.23	0.19	0.31	0.34	0.18	0.61
	45.03	50.09	30.04	38.53	39.19	37.41
LPM	0.19	0.16	0.26	0.33	0.18	0.58
	42.78	47.21	30.02	43.14	38.80	49.53
SSPM	0.07	0.03	0.17	0.17	0.05	0.38
	12.57	6.63	20.59	18.25	9.50	27.16

Table 1.2: Fit statistics. For each model, we report the mean absolute percentage error (MAPE, first row), and the root mean squared error (RMSE, second row) for both games in the forecasting task. We compute these measures over the entire range of data (Overall), over just the in-sample portion of the data (In-sample), and in just the 30 day holdout period (Holdout). Note that both of the log-logistic models were estimated over the full range of the data; given the poor fit using the full data, we did not estimate them separately using held out data.

the Log-Logistic Covs model includes the covariates as indicator variables. While this is a very common approach for specifying events of interest, as we saw in our analyses of calendar time events, the impacts of these events are unlikely to be constant over time, a fact the GPPM implicitly incorporates in the calendar time effects.

Of primary interest to us is the comparison with the customer base analysis models. We see that the fit statistics of the Pareto-NBD and BGNBD are much better than that of the hazard models. In fact, the fit of the Pareto-NBD in Figure 1.10 is similar to the calendar time muted fit in Figure 1.6. This supports our intuition that the GPPM in a sense generalizes these models, by accounting for interpurchase and lifetime effects (in a nonparametric way), while simultaneously allowing for variability in calendar time. Accounting for variability in calendar time

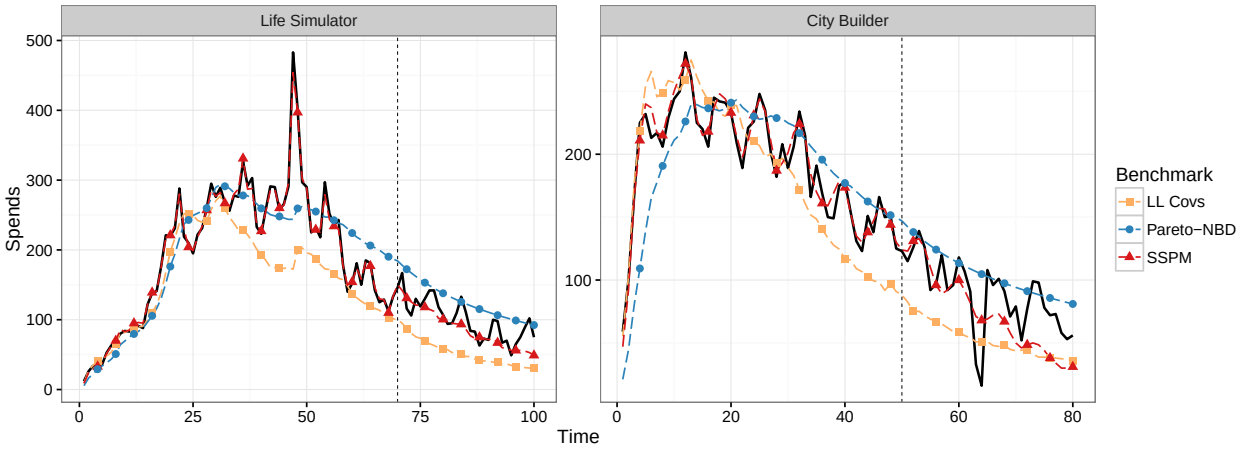

Figure 1.10: Daily spending forecasts for several of our benchmark models. The data is in black. The holdout period is the last 30 days of data, demarcated by the dashed line. A web app where all benchmark fits can be viewed in isolation and in comparison with the GPPM is available at `https://dr19.shinyapps.io/gppm_benchmarks/`.

is important, as it lets the GPPM isolate predictable individual-level effects from the influence of calendar time events. In models that rely only on recency and frequency data, calendar time events are conflated with base purchasing rates, leading to erroneous predictions in the presence of calendar time dynamics. We show this through a set of simulations in Web Appendix B.

Finally, we see that while a linear specification of the dynamic effects is clearly not sufficiently rich, resulting in the poor fit of the LPM in both settings, a non-GP nonparametric specification like in the SSPM performs similarly to the GPPM. Specifically, we see that the SSPM performs as well as the GPPM in LS, while worse than the GPPM in CB. In some sense, this is not surprising: the SSPM is a complex and novel benchmark, constructed to be equivalent to the GPPM in terms of which effects it represents and how these are modeled. Both models capture the same set of predictable individual-level and periodic calendar time effects. Forecasting spending in the GPPM relies on forecasting these propensity functions, something which the SSPM also appears to do well.[11] Unlike the GPPM,

[11]In fact, recent research has established deep links between GPs and state space models, such

however, the SSPM is more limited in its ability to separate out effects along a given time scale, which constrains its ability to perform the calendar time decompositions that are possible with GPs. This limits the SSPM's ability to provide equivalent dashboard-like representations of spend propensity along a given scale, which is one of the GPPM's core strengths.

1.4 Extensions: Simulation Studies

In this section, we use simulated data to explore two aspects of the GPPM in more depth: its extensibility via the modularity of the kernel, and its relationship to the prior literature. More specifically, in the first section below, we show how the GPPM can be extended to accommodate different length-scales of variation along different time dimensions, to capture things like "loyalty" promotions, for example, that might occur along the lifetime dimension. In the second section, we explore links between the GPPM and classic buy-till-you-die (BTYD) models for customer base analysis, focusing on the BGNBD model as our example. BTYD models have served as the backbone for many customer base analysis applications, showing a particularly robust ability to forecast future spending and compute customer-centric quantities of interest by modeling just interpurchase times and customer lifetimes. The GPPM extends this framework by also allowing for the consideration of an additional input, calendar time. To explore how the GPPM generalizes these ideas, we simulate data from both models, and show first how the recency and lifetime components of the GPPM are able to capture the equivalent BTYD effects, and second why the inclusion of calendar time effects is important in accurately estimating individual-level spend rates. Since we use simulated data across all of

that some GP models can be approximated by state-space specifications (Gilboa et al., 2015). This may also explain their similar performance.

these studies, we can see throughout examples of how the GPPM can capture the shape of events of interest automatically, as we know in these cases exactly the impact a given event.

Extending the GPPM through Kernel Modularity

Previously, we described the modular approach to specifying the GPPM. Recall that each kernel represents a broad type of functions. In the main paper, we used SE kernels to pick up variation along two length-scales, short and long, for the calendar time effects, along with a predictable periodic component. We used a single SE kernel with a monotonic power mean function to isolate variability along the other dimensions. In practice, we may want to extend the model in various ways. One potential deviation from the general model explained in the body of the paper is the need to capture short-run effects of interest that may occur along other dimensions, particularly along the lifetime dimension. These effects could exist, for instance, if the company has loyalty based rewards or promotions, such that the consumer is given a special after a certain number of days after first purchase.

To cope with shocks along the lifetime dimension, we can extend the GPPM quite simply by adding an additional SE component to the lifetime specification. By the additive property of GPs, this specification remains a GP, just with an additive kernel. Hence, we now model:

$$\alpha_{\mathrm{L}}(\ell) = \alpha_{\mathrm{L}}^{\mathrm{Long}}(\ell) + \alpha_{\mathrm{L}}^{\mathrm{Short}}(\ell),$$

where:

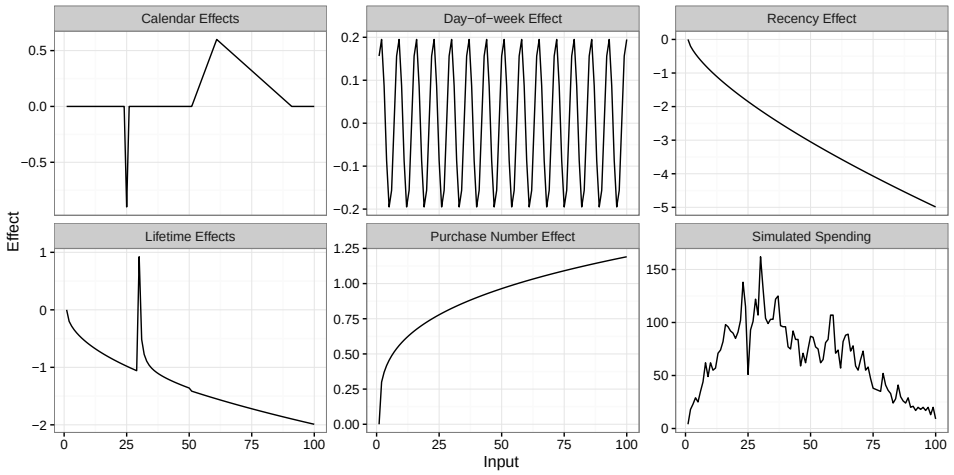

Figure 1.11: True effects used to generate the simulated data, with the simulated spending time series shown in the bottom right panel.

$$\alpha_{\mathrm{L}}^{\mathrm{Long}}(\ell) \sim \mathcal{GP}(m(\ell), k_{\mathrm{SE}}(\ell, \ell'; \eta_{\mathrm{LL}}, \rho_{\mathrm{LL}})),$$

$$\alpha_{\mathrm{L}}^{\mathrm{Short}}(\ell) \sim \mathcal{GP}(0, k_{\mathrm{SE}}(\ell, \ell'; \eta_{\mathrm{LS}}, \rho_{\mathrm{LS}})),$$

$$\rho_{\mathrm{LS}} < \rho_{\mathrm{LL}}$$

With this setup, we can capture short-run departures from the smooth trend component along the lifetime dimension, just like we captured both trends and short-run shocks in the calendar time component before. In this case, we include the same mean function as before (power mean) along the long-run curve.[12]

Simulation We simulated data within the GPPM framework, similar to the data from our application. We simulated the spending of 2,000 customers, entering over a period of 30 days, using the effects displayed in Figure 1.11. The sum of these effects results in the spending time series displayed in the bottom right panel of

[12]By additivity, the results would be equivalent if the mean function were included in the short-run term; however, we find the idea of a trend + shock formulation more intuitive, and hence model it as such.

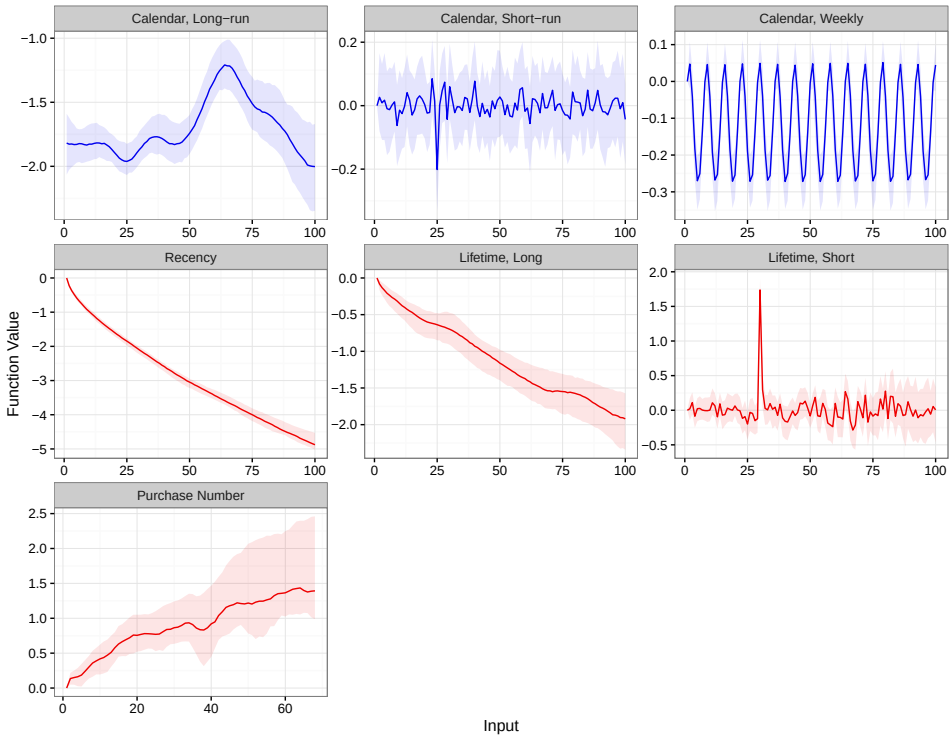

Figure 1.12: Extended GPPM dashboard on simulated data, including short and long-run components for the lifetime function.

Figure 1.11. We then estimated the GPPM on this data, using the extension described above. The resulting extended dashboard is shown in Figure 1.12. We see that the GPPM recovers all of the effects in the data generating process, without specifying any of them as inputs to the model. More importantly, we see the natural extension of the GPPM to capture the shock to the lifetime dimension. The instantaneous effect of the loyalty reward is captured in the Lifetime, Short panel, with the residual effect slight, but noticeable in the Lifetime, Long panel.

Links between GPPM and BTYD

The GPPM provides a natural generalization of buy-till-you-die customer base analysis models that rely solely on recency and lifetime, such as the BGNBD. While the GPPM does not explicitly account for customer death, it does so asymptotically

by allowing the probability of purchase to go to zero via the lifetime and recency effects. To explore this link deeper, we ran a series of simulation studies, testing in which cases the GPPM is able to capture BGNBD data, and vice versa.

We hypothesize that the dynamic spending patterns that are captured by the BGNBD can also be captured by the GPPM; however, the BGNBD will have a difficult time fitting data generated by the GPPM, depending on the strength of calendar time effects present. This is because the BGNBD and other parametric probability models based on individual-level effects have no way of separating out temporary shifts in spend propensity due to calendar time effects from underlying, predictable individual-level effects. To test these two hypotheses, we first see how the GPPM does at fitting data generated by the BGNBD model. Then we do the reverse and estimate the BGNBD on data from GPPM specifications that vary the strength and nature of the calendar time effects.

BGNBD Data, GPPM Fit If the recency and lifetime components of the GPPM do capture the dynamic patterns inherent in the BGNBD, then the GPPM should be able to do well on data generated from the BGNBD. To see this, we generate data from 8,000 spenders across 30 first spend dates, similar to our real data. We simulate spending over 100 days according to a BGNBD model, and then fit the GPPM on the first 50 days of simulated data, and forecast the activity on days 51 to 100. As our main example, we use the estimated BGNBD parameters ($r = 0.243$, $\alpha = 4.414$, $a = 0.793$, $b = 2.426$) from the original BGNBD paper (Fader, Hardie, and Lee, 2010, subsequently FHL). We also used many combinations of randomly generated parameters to test robustness, with smaller sample sizes of 2,000 customers. The fit statistics for all of the simulations are summarized in Table 1.3. The good fit offers substantial evidence to our claim that the GPPM nests these

DGP	Model	Overall	Training	Holdout
BGNBD, FHL Parameters	GPPM	0.07	0.05	0.09
BGNBD, Random*	GPPM	0.10	0.06	0.14
GPPM, All*	BGNBD	0.54	0.21	0.87
GPPM, `Nocal` Only*	BGNBD	0.22	0.15	0.29

Table 1.3: Fit summaries for the simulation studies. The first column contains the data generating process, while the second contains the model used to forecast spending. An asterisk (*) is used to denote the statistics that are the average value across many simulations. The statistics presented are MAPE (mean absolute percentage error). RMSE is not relevant here as each simulation results in spending on a different scale, and hence RMSE is not comparable across simulations.

traditional probability models.

GPPM Data, BGNBD Fit We also study the reverse situation and examine the performance of the BGNBD on data generated from the GPPM. We show that BGNBD is not able to fit such data very well, especially in the presence of calendar time dynamics. Specifically, we use three levels of the day of the week effect — none (`Nocyc`), weak (`Weakcyc`), and strong (`Strongcyc`) — and three kinds of non-cyclic calendar time effects: none (`Nocal`), a long-run peak similar to the general holiday season bump seen in our application (`Peakcal`), and a nonlinear decreasing trend across the whole time period (`NonlinDeccal`). The cyclic effect was set as $\alpha_w(t) = \theta \sin(2\pi t/7)$, where $\theta = 0$, for no cyclic effect, $\theta = 0.15$, for the weak effect, and $\theta = 0.4$, for the strong effect. For the calendar time effects, the non-linear decreasing calendar time trend is given by $\alpha_{\mathrm{T}}(t) = -0.2t^{0.3}$; the peak effect is given by the piecewise function: $\alpha_{\mathrm{T}}(t) = 0$, when $t \leq 20$; $\alpha_{\mathrm{T}}(t) = 0.5(t - 20)$, when $t \in [21, 40]$; $\alpha_{\mathrm{T}}(t) = 0.1(50 - t)$, when $t \in [41, 50]$ and $\alpha_{\mathrm{T}}(t) = 0$, when $t > 50$.

Figure 1.13 and Table 1.3 show the results from these simulations. We see that BGNBD fits the mean of the curve in the presence of a cyclic effect. We also see that the BGNBD generally does well in the cases where there is no short or long-run calendar variation, underpredicts in the beginning and then overpredicts in

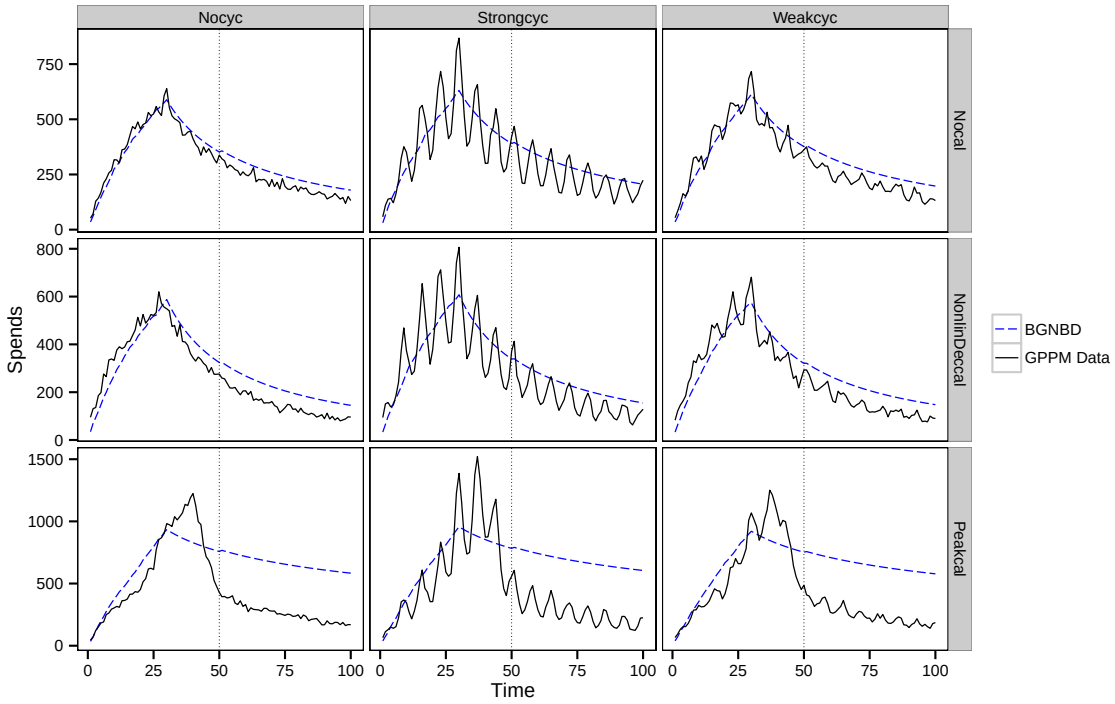

Figure 1.13: The BGNBD fit on various types of data drawn from the GPPM: `Nocyc`, `Strongcyc`, and `Weakcyc` indicate no, strong, and weak cyclic (day of the week) effects respectively; `Nocal` indicates no calendar time dynamics, `NonlinDeccal` indicates a non-linear decreasing long-run calendar time process, and `Peakcal` indicates a calendar time process that is flat but with a peak during the calibration period.

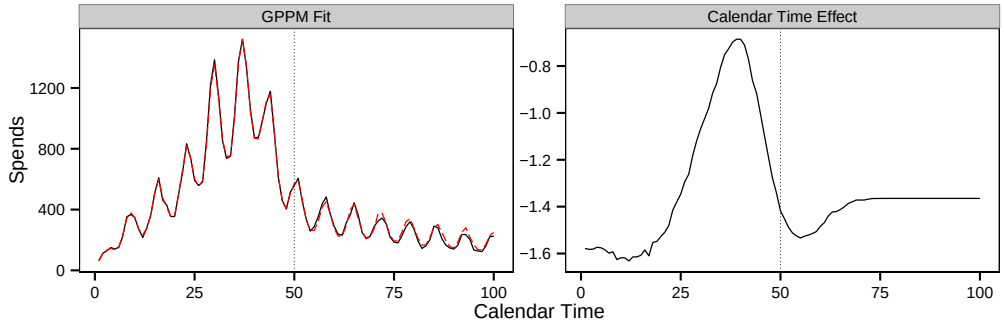

Figure 1.14: The GPPM fit and forecast on the `Strongcyc/Peakcal` simulated data, together with the estimated calendar time effect. We see that the GPPM captures the pointed piecewise effect, and is therefore able to isolate the predictable, individual-level effects that allow it to accurately forecast future spending.

the end when there is a decreasing calendar time effect, and fails significantly at capturing the peak effect. We see in the `Peakcal` case (last row of Figure 1.13) that the BGNBD attributes the peak to higher rates of spending, and then dramatically

overestimates future spending.

The GPPM does not fall prey to this same bias because of its ability to separate out calendar time effects. To emphasize this, we see the GPPM fit to the worst case (`Strongcyc/Peakcal`), together with the estimated calendar time effect, in Figure 1.14. The excellent fit and near perfect forecast is not surprising: the GPPM is capturing data generated from a GPPM. One thing to point out is that this, again, demonstrates the ability of the GPPM to nonparametrically recover the effects of events, as we see the peak in calendar time is equivalent to the piecewise function described above.

1.5 Conclusion

In this paper, we developed a highly flexible model-based approach for understanding and predicting spending dynamics. Our model, the Gaussian process propensity model, or GPPM, employs Bayesian nonparametric Gaussian process priors to decompose a latent spend propensity into components that vary along calendar time, interpurchase time, customer lifetime, and purchase number dimensions. Our additive structure yields easily interpretable model outputs and fits customer spending data well.

We showed that the GPPM identifies the latent dynamic patterns in the data via a principled probabilistic framework that reliably separates signal from noise. It offers a number of outputs that are of considerable value to managers. First, the GPPM generates a dashboard of latent functions that characterize the spending process. These model-based dashboards are easy to comprehend, even by managers who may lack sophisticated statistical skills. Second, we demonstrated

that the GPPM is capable of automatically capturing the effect of events that may be of interest to managers. In situations where certain events may escape the notice of managers, the GPPM is able to detect these events automatically. More importantly, the nonparametric nature of the GPPM allows it to flexibly model the nature and duration of the impact of events (either known or unknown, a priori), without the need to represent these explicitly via covariates. These advantages of the GPPM make it ideal for decision contexts involving multiple products and information asymmetries. The GPPM also flexibly captures the individual-level drivers of spending that reliably explain and predict spending behavior, including recency, lifetime, and purchase number effects. These effects can be used to characterize spending patterns within distinct customer bases, analyze individual customer respend probabilities, and predict future spending activity. Furthermore, since these effects are estimated jointly with the calendar time events, as part of a unified propensity model, the predictable, fundamental individual-level drivers of spending are determined net of potentially unpredictable calendar time effects. Moreover, calendar time events can be analyzed net of the impact of expected individual-level spend activity, in a way not possible with mere aggregate data analysis.

We demonstrated these benefits of the GPPM on two data sets of purchasing activity within mobile games. We illustrated how the model-based dashboards that are generated from the GPPM yield easily interpretable insights about fundamental patterns in purchasing behavior. We also showed that the GPPM outperforms traditional customer base analysis models in terms of predictive performance, both in-sample and out-of-sample, including hazard models with time-varying covariates and the class of buy-till-you-die models. The predictive superiority of the GPPM stems from the fact that it captures the same predictable effects as traditional customer base analysis models, like recency and lifetime, but does so in a flexible

way, net of the influence of calendar time events.

While the paper showcases the many benefits of our framework, it is also important to acknowledge some limitations. First, the framework in its current form is computationally demanding, especially when compared with simpler probability models that can be estimated with maximum likelihood. It is also data intensive. In our application, we used complete individual-level event log data to estimate the model. Some of the benchmark models, in particular, the BGNBD and the Pareto-NBD, use only two sufficient statistics per customer. Both of these limitations can perhaps be addressed in practice by either data subsampling, or by developing faster inference algorithms. Finally, while we believe our model-based dashboard is useful, insofar as it provides a snapshot of the key drivers of spending dynamics, it does not work in real-time, as is the case for many dashboards of marketing metrics. A streaming data version of our model would be an interesting area for future work.

To conclude, we believe the GPPM addresses a fundamental need of modern marketing managers for a flexible system for dynamic customer base analysis. In providing a solution to this problem, this work introduces a new Bayesian nonparametric approach to the marketing literature. While we discuss Gaussian Process priors in the context of dynamic customer base analysis, their potential applicability to other areas of marketing is much broader. GPs provide a general mechanism for flexibly modeling unknown functions, and for doing Bayesian time series analysis. We see many potential applications for GPs in marketing, including in the modeling of the impact of marketing mix variables, such as advertising and promotions, and in the approximation of unknown functions in dynamic programming and other simulation contexts. Our work also makes a contribution to the largely unaddressed field of visual marketing analytics systems, or dashboards.

Dashboards and marketing analytics systems are likely to become even more important in the future, given the increasing complexity of modern data-rich environments. As dashboards increase in relevance, we believe that managers will welcome further academic research in this domain.

2

Dynamic Preference Heterogeneity

This essay forms the basis of a paper, "Dynamic Preference Heterogeneity," which is currently under review at the Journal of Marketing Research. That paper is jointly authored with Asim Ansari and Yang Li.

Abstract

Consumers' preferences and sensitivities to marketing variables change over time, often in tandem with population trends, but frequently exhibiting individual-specific idiosyncrasies. While much of the empirical marketing literature has focused on capturing cross-sectional heterogeneity, little research has been done on modeling the temporal evolution of heterogeneity. In this work, we develop a Bayesian nonparametric framework based on Doubly Hierarchical Gaussian Processes (DHGP) for modeling dynamic heterogeneity, which flexibly captures both the evolution of population trends and individual-level departures from those trends over time. This novel specification allows for sharing of statistical information across individuals, and within individuals over time, to provide rich individual-level insights and efficient inferences regarding dynamics. We showcase our DHGP specification in a choice modeling context, using both simulations and an application to consumer packaged goods data. We find that restricted heterogeneity specifications, as have been employed previously in the literature, can lead to significant biases in the presence of dynamic heterogeneity, even in estimating population-level trends. Moreover, these restricted specifications cannot capture managerially-relevant patterns of individual-level variation around population trends. In our application, we show robust evidence of dynamic heterogeneity across CPG categories during the Great Recession, and illustrate the clear gains from capturing dynamic heterogeneity through our DHGP specification. We uncover important individual-specific trends that can be used for targeting, including variability in consumer responses to the recession, and show that targeted pricing that leverages dynamic heterogeneity can lead to higher retailer profits.

2.1 Introduction

Marketers have long appreciated the fact that customers differ in their preferences for products and services, and that customer preferences are dynamic in nature. Individual preferences are shaped by a number of factors that stem from past consumption episodes and personal experiences. As these experiences typically vary across consumers, they result in differences in tastes for attributes and brands. Preferences and response sensitivities of consumers are also influenced by economic conditions and the advertising and promotional activities of firms. Shocks to incomes and budget constraints emanating from economic downturns can impact price and promotion sensitivities of consumers, as well as shift preferences toward more economical brands. Preferences can also vary over time because of changes in tastes for certain attributes that reflect broad societal trends, such as an increasing health consciousness in society resulting in increased liking for healthy products. Preferences, therefore, are necessarily both heterogeneous and dynamic in nature.

Importantly, preferences are also heterogeneous in their dynamics. While economic shocks or societal trends may induce common patterns in consumers' response sensitivities—for instance, increasing price sensitivity during a recession—consumers are affected by these factors to varying degrees. Preferences also change because of consumption feedback, learning, and variations in information sets, all of which are inherently individual-specific phenomena. Capturing the differences in individual-level evolution of preferences, which we call dynamic heterogeneity, is thus important to gain a proper understanding of the drivers of customers' choices, and is the focus of our paper.

Many different forms of heterogeneity have been modeled in the marketing literature. DeSarbo et al. (1997) in their review paper distinguish among response,

structural, perceptual, form, distributional, and time heterogeneity. Much of the literature focuses on modeling variation in preferences across individuals, but variation within individuals over time has been relatively understudied. However, modeling this intra-individual variation has important managerial implications for understanding changes in markets over time, and for developing dynamic and forward-looking segmentation and targeted pricing strategies. In addition, just as ignoring cross-sectional heterogeneity can result in misleading inferences about response sensitivities, not accounting for parameter evolution can also distort inferences and misinform managerial actions.

In this paper, we develop a novel Bayesian nonparametric approach for dynamic heterogeneity: the Doubly Hierarchical Gaussian process (DHGP). We embed DHGP in a discrete choice model, fusing standard marketing models of choice and modern machine learning methods for flexible functional modeling (Rasmussen and Williams, 2006; Roberts et al., 2013; Dew and Ansari, 2016). The DHGP dynamic heterogeneity specification allows us to capture both population trends in preferences over time, and individual-level departures from those trends. It also hierarchically estimates the hyperparameters of this form of heterogeneity in a fully Bayesian fashion. This doubly hierarchical specification allows for sharing statistical strength both across individuals at any given point in time, and within individuals across time periods. Our DHGP approach to nonparametrically modeling coefficients in a latent variable model is novel to the econometric, marketing, and machine learning literatures. While we showcase the value of our framework within the context of choice models, capturing the evolution of individual-level model parameters is relevant in many marketing, psychometric, and analytics settings (Liechty et al., 2005; Bockenholt, 2006; Wedel and Kannan, 2016), and our dynamic heterogeneity framework can be easily adapted to those contexts.

Given the importance of modeling evolving preferences, some researchers have extended standard choice models to allow for time-varying parameters. Often, these models focus on capturing specific mechanisms of preference evolution. For instance, Guadagni and Little (1983) model both the heterogeneity and the evolution of brand preferences using exponentially smoothed customer-level brand-loyalty parameters. Alternatively, the mechanism of consumer learning can be modeled explicitly, as in Roberts and Urban (1988). Others have modeled preference parameters as functions of marketing actions using distributed lags (Mela et al., 1997; Seetharaman and Chintagunta, 2003).

In contrast, more recent work has focused on capturing general parametric evolution in choice models. The most common heterogeneity specification in these models is what we term the fixed-offsets (FO) specification, wherein individuals are allowed to differ in their preferences, but where the time evolution of those preference is restricted to move in parallel to the population trend (Neelamegham and Chintagunta, 2004; Liechty et al., 2005; Kim et al., 2005; Lachaab et al., 2006; Sriram et al., 2006; Sriram and Kalwani, 2007; Guhl et al., 2018). Such a restricted specification is clearly unrealistic, as the underlying mechanisms driving preference dynamics operate at an individual-level, and hence preference dynamics should be able to vary flexibly at the individual-level. Other work avoids such restrictions by simply modeling time periods independently (e.g. Gordon et al., 2013). This independent-periods (IP) approach is fully flexible, and will not lead to biases. However, it is generally inefficient, as it ignores the fact that individual-level parameters necessarily evolve gradually, and that parameters in adjacent time periods tend to be similar.

In this work, we bridge the fixed-offsets (FO) and independent-periods (IP) approaches by specifying a random utility model that incorporates both global and

individual-level parameter dynamics in a flexible, yet principled fashion. In particular, we develop a class of Doubly Hierarchical Gaussian Process (DHGP) models that allows for information sharing both across individuals, as in the classic hierarchical choice models, and across time periods. This is a novel dynamic heterogeneity specification as it models differences in functions of time (i.e., stochastic processes), rather than differences in scalar parameters. This approach allows for flexible modeling of both global and individual-level patterns, nests existing models as special cases, and efficiently uses information across consumers as well as time periods. As one of the first papers in marketing to make use of the powerful Bayesian nonparametric Gaussian process methodology, our work also contributes to an important and growing stream of non- and semiparametric models in marketing, which offer data-driven insights to managers with minimal modeling or structural assumptions (Kim et al., 2004; Ansari and Iyengar, 2006; Rossi, 2013; Li and Ansari, 2014; Dew and Ansari, 2018).

We illustrate the benefits of our modeling framework using both simulated data and real panel data on consumer choices in two popular consumer packaged goods categories. We compare inferences from models estimated with our DHGP dynamic heterogeneity specification to the FO and IP approaches. Using simulations, we show that, in the presence of dynamic heterogeneity, the popular FO specification can lead to biased population estimates and an understimation of the magnitude of heterogeneity in the population. We also show the inefficiencies that come from ignoring the dependencies between adjacent time periods, as in the IP approach. In our application, we show that the nuanced individual-level dynamics that can be recovered through a dynamic heterogeneity specification are lost when using restricted or inefficient models. However, these dynamics have important managerial implications. Specifically, using panel data of spending during the Great Recession, we show a clear impact of the recession on consumers' price sensitivities,

including interesting patterns of intra-individual variation. Importantly, we also uncover significant time variation in the base preferences for different brands. Managerially, we show that this flexible handling of individual-level dynamics yields important insights, and reveals a significant number of consumers whose brand preferences are evolving counter to the population trajectory. We also find that alternative approaches may underestimate the gains from targeted pricing, as compared to those from a pricing strategy that leverages dynamic heterogeneity.

The rest of the paper is organized as follows: in Section 2.2, we give an overview of our modeling context, and describe existing approaches to modeling preference evolution. We then give a primer on Gaussian processes in general, and a detailed description of our doubly hierarchical Gaussian process dynamic heterogeneity specification. In Section 2.3, we establish the relative merits of our model on synthetic data. In Section 2.4, we describe our data, summarize our results, and explore the managerial implications of accounting for dynamic heterogeneity, culminating in an application to optimal discounting. Finally, in Section 2.5, we summarize our findings, cite some limitations of the current work, and suggest areas for future research.

2.2 Modeling Framework

We model heterogeneous time-varying preferences within the standard set up of random utility discrete choice models. We index consumers by $i = 1, \ldots, I$ and choice alternatives by $j = 1, \ldots, J$. In specifying the time variation in preferences, we distinguish between calendar time periods, indexed by $t = 1, \ldots, T$, and consumer-specific choice occasions, indexed by m. As a consumer can have zero or more choice occasions within a particular calendar time period t, we use $t_{(i,m)}$ to

denote the calendar time period associated with the mth observation of individual i. The utility function for a choice alternative j, for consumer i on choice occasion m can then be written as

$$u_{ijm} = \mathbf{x}_{ijm}^{\top} \boldsymbol{\beta}_{i,t_{(i,m)}} + \epsilon_{ijm}, \tag{2.1}$$

where $\mathbf{x}_{ijk}$ is a vector of observed explanatory variables (including brand-specific dummies) faced by the consumer on that choice occasion and $\boldsymbol{\beta}_{i,t_{(i,m)}}$ is the corresponding vector of preference coefficients.[1] The stochastic component of the utility ϵ_{ijm} is assumed to be distributed i.i.d. extreme value, across brands and observations. With the standard assumption that a consumer chooses the alternative j with the highest utility (i.e., $u_{ijm} > u_{ilm}, \forall\, l \neq j$), choice probabilities are given by the familiar logit choice formula,

$$P_{ijm} = \frac{\exp(\mathbf{x}_{ijm}^{\top} \boldsymbol{\beta}_{i,t_{(im)}})}{\sum_l \exp(\mathbf{x}_{ilm}^{\top} \boldsymbol{\beta}_{i,t_{(im)}})}. \tag{2.2}$$

We model the time variation in preferences by assuming that the parameter vector for a consumer varies across the calendar time periods, but remains the same for all observations within a time-period. In other words, we assume that all observations for consumer i within a given time period t share the same preference vector $\boldsymbol{\beta}_{it}$, (i.e., $\boldsymbol{\beta}_{i,t_{(i,m)}} = \boldsymbol{\beta}_{it}$, when $t_{(i,m)} = t$). This allows us to align the preference parameters of different consumers onto a common time scale. Before we present our Bayesian nonparametric framework to model dynamic heterogeneity, we briefly describe other approaches to modeling temporal evolution of parameters.

[1] For this paper, the $^{\top}$ symbol will denote transposition, while the $'$ symbol will denote distinct elements, as in, for example, two inputs to a time-varying function being denoted t and t'.

2.2.1 Existing Models of Preference Evolution

Researchers have predominantly used two approaches to capture the temporal evolution of model coefficients, $\boldsymbol{\beta}_{it} = (\beta_{it1}, \ldots, \beta_{itP})$. The simplest option, which we call the independent-periods (IP) approach, assumes that an individual consumer draws a new vector of coefficients every time period. In particular, the individual-level coefficients in a time-period t are assumed to come from a period-specific population distribution,

$$\boldsymbol{\beta}_{it} \overset{iid}{\sim} \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Lambda}), \tag{2.3}$$

where $\boldsymbol{\mu}_t$ is the population mean for period t and $\boldsymbol{\Lambda}$ is a population covariance matrix. In this specification, the only link between the population distributions comes from the common $\boldsymbol{\Lambda}$ across time periods, although this restriction can also be relaxed. While this approach is very flexible, in the sense that it can capture any temporal patterns in sensitivities when sufficient data are available, it is not efficient, as it does not utilize the fact that nearby time periods are more likely to be related than distant periods, or that an individual's preferences can be more highly correlated over time, relative to the population. Moreover, this approach does not permit a natural mechanism for forecasting, as the nature of the evolution of parameters is not explicitly modeled. Without additional assumptions, this prevents such a specification from being used in predicting demand, or setting optimal prices or targeted strategies in future periods. As such, we will consider comparisons to this method in the simulation studies, but we will focus in our applications on modeling parametric evolution.

The most popular approach in marketing to specifying heterogeneity when modeling parametric evolution is to simply assume static heterogeneity. That is, to

model dynamics in preferences at the population-level via a time series specification that links the population mean from one period to the next, and assume that the pattern of variation at the consumer-level exactly mimics the population dynamics (Neelamegham and Chintagunta, 2004; Liechty et al., 2005; Kim et al., 2005; Lachaab et al., 2006; Sriram et al., 2006; Sriram and Kalwani, 2007; Guhl et al., 2018). In these models, consumer-level trajectories are assumed to deviate from the population-level trajectory by an individual-specific fixed (i.e., time-invariant) offset. Hence, we often refer to this model as the fixed offsets (FO) model. Conditioned on the path for the population mean, $\{\boldsymbol{\mu}_t\}_{t=1}^{T}$, the consumer-specific sensitivities are given by

$$\begin{aligned} \boldsymbol{\beta}_{it} &= \boldsymbol{\mu}_t + \boldsymbol{\delta}_i, \\ \boldsymbol{\delta}_i &= (\delta_{i1}, \ldots, \delta_{iP}) \sim \mathcal{N}(0, \boldsymbol{\Lambda}) \,. \end{aligned} \tag{2.4}$$

Under this specification, the mean shifts are modeled from time period to time period, but heterogeneity around the mean is fixed. The coefficients for consumer i are always a fixed offset distance $\boldsymbol{\delta}_i$ from the population mean $\boldsymbol{\mu}_t$, in every time period. Thus, while individuals are allowed to differ from each other in their sensitivity to a given marketing variable, the way a consumer's sensitivities vary over time is determined purely via the population dynamics.

Apart from the above, some researchers have used state-space formulations in which the consumer-level parameters are allowed to evolve independently from a fixed initial prior distribution, as in DeSarbo et al. (2005). While this specification offers considerable flexibility in capturing different consumer-level parameter evolution patterns, it allows for pooling of the individual-level trajectories only in the initial period, and thus is ill-suited for observational data where individuals do not purchase consistently over time. Moreover, this strategy does not yield a ready

estimate of the population-level trajectory, something that is valuable for understanding the aggregate patterns of dynamics.

In this paper, we address the shortcomings of previous approaches for estimating preference evolution at the individual-level by treating the individual-level coefficients $\boldsymbol{\beta}_{it}$ as *functions* of time t, rather than as period-specific parameters. We then model these functions nonparametrically using Gaussian process (GP) priors. These priors allow us to model individuals as nonparametrically deviating from a mean function, $\boldsymbol{\mu}_t$, which represents the population trajectory, and can be estimated either through another GP, or an alternate time series or state space model. The use of GPs in this way offers a flexible, nonparametric mechanism for specifying preference evolution at the individual-level that pools information both across consumers and across time periods. These traits allow our doubly hierarchical GP specification to capture the evolution of heterogeneity over time.

Although GPs have been explored extensively for functional modeling within statistics and machine learning, they have received limited attention in marketing. Moreover, to the best of our knowledge, they have not been used previously in specifying parameter-driven dynamics within discrete choice models. Given their novelty to the marketing audience, we give a brief overview of GPs before discussing how they are used in our framework. For a full treatment of GPs, we refer the reader to Rasmussen and Williams (2006). For a more extensive overview than the one below, and for an application of GPs in a marketing context, see the previous chapter of this dissertation.

2.2.2 Gaussian Processes Redux

Recall from Essay 1 that a Gaussian process is a stochastic process f over some input space, which in the present work, we take to be time, $t \in \mathbb{R}$. GPs are defined by a mean function, $m(t)$, and a covariance function or kernel, $k(t, t')$ over input pairs (t, t') such that $m(t) = \mathbb{E}[f(t)]$, and $k(t, t') = \text{cov}(f(t), f(t'))$. If $f \sim \mathcal{GP}(.)$, then for any finite set of inputs, $\boldsymbol{t} = (t_1, \ldots, t_T)$, the collection of corresponding function values over these inputs has a joint multivariate Gaussian distribution,

$$f(\boldsymbol{t}) = (f(t_1), \ldots, f(t_T)) \sim \mathcal{N}(m(\boldsymbol{t}), K(\boldsymbol{t}))\,, \tag{2.5}$$

where $m(\boldsymbol{t}) = (m(t_1), \ldots, m(t_T))$ is the mean vector of the multivariate normal and K is the $T \times T$ covariance matrix with entries given by $K_{ij} = k(t_i, t_j)$. This capacity to specify a distribution over outputs for any given set of inputs means GPs provide a natural mechanism for specifying uncertainty over a function space. In our context, we treat the model coefficients to be functions of time and use GPs to specify the temporal variation in parameters.

The choice of the mean function and the kernel determines the nature of the functions that a GP prior generates. Informally, the mean function encodes the expected location of the functions, whereas the kernel encodes function properties, such as smoothness, amplitude, and differentiability. In much of the GP literature, the mean function is chosen to be constant, to reflect a lack of prior assumptions about the shapes of the functions, and the kernel serves as the source of model specification.

Kernel Choice A number of different kernels have been proposed in the wider GP literature. In this work, we rely primarly on the rich class of Matérn kernels, which

has a general form given by:

$$k(t, t'; \eta, \kappa, \nu) = \eta^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\kappa \left|t - t'\right|\right)^{\nu} K_{\nu}(\kappa \left|t - t'\right|), \tag{2.6}$$

where $\eta > 0$, $\kappa > 0$, and $\nu > 0$ are the kernel hyperparameters that govern the characteristics of the function draws, $\Gamma(\cdot)$ is the gamma function, and $K_{\nu}(\cdot)$ is the modified Bessel function of the second kind. While the functional form of the kernel is unintuitive, its hyperparameters have straightforward meanings: the amplitude η controls the variability of the function draws around the mean function, while κ, the inverse length-scale, determines the smoothness of those function draws.[2] The degree ν also determines the smoothness of the functions, by determining the level of differentiability of the function draws, as draws from a GP with a Matérn kernel are $\lceil \nu - 1 \rceil$ times differentiable, where $\lceil \cdot \rceil$ is the ceiling function. In the context of dynamic heterogeneity, η determines the magnitude of dynamic heterogeneity, as it reflects how far individual-level curves fall from the mean curve, while κ captures the degree of intra-individual intertemporal pooling, which we will elaborate more on below.

The finite differentiability property of the Matérn kernel means that the function draws can exhibit "wiggly" behavior, which is ideally suited for temporal preference data, as we need to allow for the possibility of momentary fluctuations in observed sensitivities, while still capturing the underlying smoothness of the process. When the degree is fixed to a half integer ($\nu = n + 1/2,\ n \in \mathbb{N}$), the forbidding

[2]Note here we use an inverse length-scale, and slightly rescaled parametrization, instead of the more typical form given by:

$$k(t, t'; \eta, \rho = 1/\kappa, \nu) = \eta^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left[\frac{\sqrt{8\nu}\left|t - t'\right|}{\rho}\right]^{\nu} K_{\nu}\left(\frac{\sqrt{8\nu}\left|t - t'\right|}{\rho}\right).$$

Our parameter $\kappa = \sqrt{8\nu}/\rho$. This follows the discussion of Fuglstad et al. (2018). Using an inverse length-scale allows us to nest the fixed offsets model as a special case, and is amenable to our choice of prior for the hyperparameters. The rescaling also helps with the interpretability of the prior.

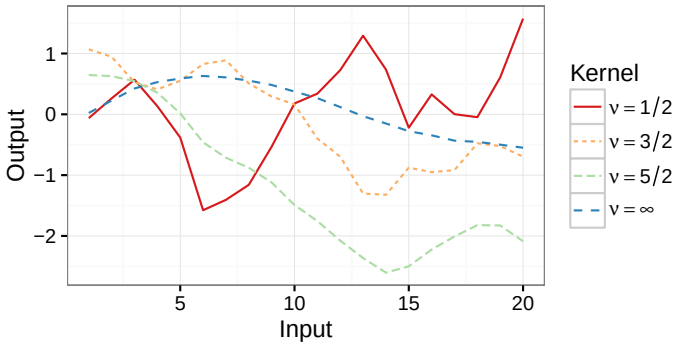

Figure 2.1: The role of the degree parameter ν in determining the smoothness of GP draws with the Matérn kernel. (A color version of this figure is available online.)

functional form in Equation 2.6 simplifies to a product of a $\lceil \nu - 1 \rceil$ degree polynomial and an exponential. Furthermore, when the degree $\nu \to \infty$, the kernel converges to the common squared exponential kernel, which allows for capturing infinitely smooth functions. We show function draws from a zero mean GP with the Matérn kernel for $\nu = 1/2, 3/2, 5/2$, and ∞, in Figure 2.1. The figure illustrates how the degree of the kernel influences the smoothness of the function draws.

Prior work has shown that the hyperparameters of the Matérn kernel cannot all be consistently estimated, and that in particular, ν cannot be separately identified from ρ (Zhang, 2004; Kaufman and Shaby, 2013). Hence, when using a Matérn kernel, ν is typically fixed to a value that reflects the supposed smoothness of the underlying process. Moreover, when the degree is fixed to a half integer ($\nu = n + 1/2,\ n \in \mathbb{N}$), the forbidding functional form in Equation 2.6 simplifies to a product of a $\lceil \nu - 1 \rceil$ degree polynomial and an exponential. For example, when $\nu = 3/2$, the kernel simplifies to:

$$k(t, t'; \eta, \kappa) = \eta^2 \left(1 + \kappa \left|t - t'\right|\right) \exp(-\kappa \left|t - t'\right|) . \tag{2.7}$$

Fixing ν to a half integer thus makes working with the kernel more tractable. Furthermore, when the degree $\nu \to \infty$, the kernel converges to the squared

exponential kernel, which is the kernel employed in Dew and Ansari (2018). In Figure 2.1, we show function draws from a zero mean GP with the Matérn kernel for $\nu = 1/2, 3/2, 5/2$, and ∞, which are the four values for ν we consider in this work. Limiting the kernel to these four degrees is in keeping with past literature, which shows little benefit for considering degrees between 5/2 and ∞ (Rasmussen and Williams, 2006).

We use the Matérn kernel class in this work for several reasons. First, the finite differentiability property of the Matérn kernel means that the function draws can exhibit "wiggly" behavior, which is ideally suited for temporal data, especially preference data (Application 1), as we need to allow for the possibility of momentary fluctuations in observed sensitivities, while still capturing the underlying smoothness of the process. Second, this class nests the squared exponential kernel as a limiting case, which allows our discussion to extend to that kernel. The squared exponential is the typical workhorse of the GP literature, and is the kernel employed by Dew and Ansari (2018). Finally, there exists a class of complexity penalizing priors for Matérn kernels, which we rely on for estimating our DHGP specification in a principled, fully Bayesian fashion, as we describe below.

We also use the fact that a GP with a non-zero constant mean function $m(t) = m$ and an arbitrary kernel $k(\cdot)$ can be equivalently represented as a GP with zero mean and an addition of a constant term ϕ^2 to the kernel $k(\cdot)$.[3] This constant additive term is often called the bias kernel. While these specifications are exactly equivalent, we have found that the bias kernel version offers greater stability in estimating these models on real data, and hence prefer it over the explicit mean function representation.

[3]Mathematically, if $f \sim \mathcal{GP}(\mu, k(\cdot))$ with a scalar constant mean μ, and $\mu \sim \mathcal{N}(0, \phi^2)$, then for fixed inputs, we have $f \sim \mathcal{N}(\mu, K)$, and we can marginalize out μ, which yields $f \sim \mathcal{N}(0, K + \phi^2 \mathbf{1}\mathbf{1}')$. In other words, $f \sim \mathcal{GP}(\mu, k)$ with a normal prior on μ is equivalent to $f \sim \mathcal{GP}(0, k + \phi^2)$.

2.2.3 Doubly Hierarchical Gaussian Process Dynamic Heterogeneity

Having established both the needed background literature on dynamic choice models and Gaussian process priors, we now return to our original goal of modeling dynamic heterogeneity. To capture consumer-level, time-varying preferences such that statistical information is shared both across time periods and across consumers, we model individual-specific parameters as functions of time. These functions are assumed to come from a GP whose mean function encodes the population-level dynamics. This specification results in a novel approach to estimating heterogeneity, which we term the Doubly Hierarchical Gaussian Process (DHGP). When employed in choice models, DHGP captures individual-specific deviations from population-level trends in a principled, probabilistic fashion. While a few researchers have employed variants of hierarchical GP models (e.g. Damianou and Lawrence, 2013; Yang et al., 2016), our approach is unique because we model the coefficients of a choice model, which are latent, rather than the response function itself.

Mathematically, our goal is to estimate a time-varying, individual-level parameter, indexed by p, denoted β_{ipt}, which represents either a consumer's sensitivity to a marketing variable, or a brand intercept in the consumers indirect utility function. We assume the researcher has a mean model of interest, μ_{pt}, which can be evaluated for any input t.[4] In this sense, we can re-write μ_{pt} as a function of time, $\mu_p(t) = \mu_{pt}$. Conditional on that model, we assume the individual-level parameter β_{ipt} is also a function of time, $\beta_{ip}(t)$. We then use Gaussian processes to

[4]This is not really a limiting assumption: a huge class of time series specifications meet this criterion.

specify a distribution over this space of individual-level functions, such that:

$$\beta_{ip}(t) \sim \mathcal{GP}(\mu_p(t), k(t, t'; \phi_p)). \tag{2.8}$$

In words, we transform the problem of estimating heterogeneity around a dynamic model into a problem of estimating individual-level functions of time centered around that model. Conditional on the mean model, we assume that individual-level departures from the mean model are governed by individual-level Gaussian processes, with a shared set of hyperparameters.

As noted before, we will rely on this work on the Matérn class of kernels with a fixed degree parameter $\nu = d$ (typically with $d = 3/2$) to specify DHGP, such that:

$$k(t, t'; \phi) = k_{\text{Mat}}(t, t'; \phi_p = \{\eta_p, \kappa_p\}, \nu = d) \tag{2.9}$$

We favor an inverse length-scale parametrization here due to the natural link between our model and the FO specification, as we describe subsequently. This yields a flexible class of choice models with dynamic heterogeneity, variants of which differ in the manner in their mean function specification. Intuitively, our hierarchical specification means that consumer-level dynamics are modeled as nonparametrically deviating from population dynamics. The traits of these individual-level curves are captured by the kernel hyperparameters η_p and κ_p. Information is shared across consumers via the common mean function $\mu_p(t)$ and the common kernel hyperparameters. The hyperparameter η_p determines the degree of inter-individual variation for a given coefficient p, by controlling how far the individual-level curves can move from the population curve. The inverse length-scale hyperparameter κ_p governs the amount of intra-individual variation, by determining the degree of correlation in an individual's function values over time,

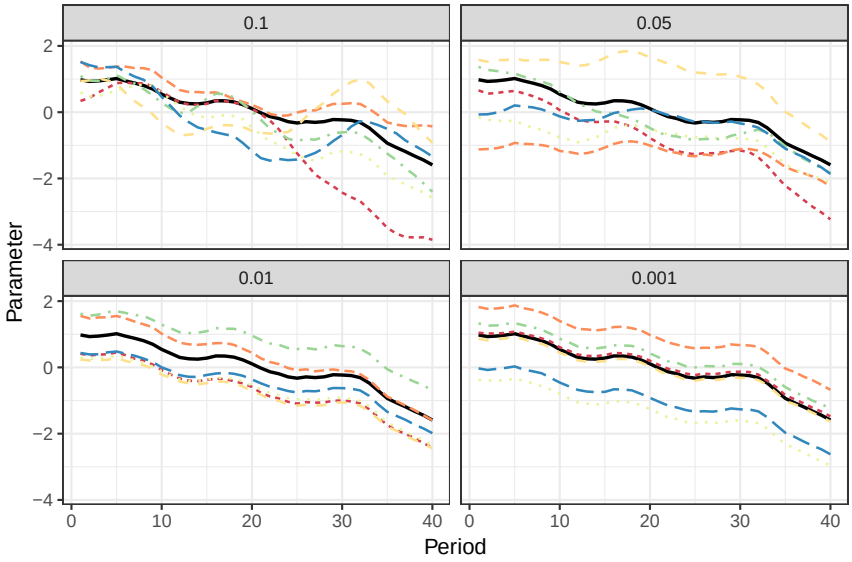

Figure 2.2: Draws from a doubly hierarchical GP across different values of the inverse length-scale, κ. The mean function is denoted by the bold solid line. (A color version of this figure is available online.)

net of the population trend $\mu_p(t)$. In other words, κ_p determines how smoothly individual's parameter functions $\beta_{ip}(t)$ deviate from the population function $\mu_p(t)$. We illustrate the role of κ_p in Figure 2.2.

Since we estimate the parameters ϕ in a fully Bayesian fashion, this is a hierarchical Gaussian process, in the sense of Flaxman et al. (2016). However, it is also hierarchical in the sense of classical hierarchical Bayes models for random effects and coefficients: each individual has his or her own parameters, which are shrunk toward population mean. In that sense, we call our specification a *doubly* hierarchical Gaussian process (DHGP). Note that GPs in this case are estimating *individual-level* functions, which is distinct from using GPs to estimate mean dynamics, as in Dew and Ansari (2018). Moreover, we are assuming that $\beta_i(t)$ is the parameter of some model, and thus a latent quantity. This is distinct from using GPs to smooth observed time series, as in Yang et al. (2016).

Hyperparameters

There are two key hyperparameters of interest when employing DHGP: the amplitude parameter η and the inverse length-scale κ. η captures the magnitude of dynamic heterogeneity—that is, how far individuals typically depart from the population mean—while κ captures the smoothness of those departures, representing the degree of intertemporal, intra-individual information sharing. As we've shown, when $\kappa \to 0$, DHGP becomes the fixed offsets specification. Thus, estimating these two parameters is of great importance.

We employ a fully Bayesian estimation strategy for estimating the DHGP hyperparameters. In particular, we leverage the Penalized Complexity (PC) prior for Matérn Gaussian random fields introduced by Fuglstad et al. (2018). The PC prior is a weakly informative prior, based on the idea of penalizing the complexity induced by the kernel hyperparameters in the resultant Gaussian process. Complexity, in the case of classical GP regression, refers to functions with high amplitude (large η) and small length-scales (small ρ, equivalent to large κ). In Fuglstad et al. (2018), the authors derive their PC prior by placing a prior on the KL divergence between the full model with unrestricted kernel hyperparameters, and the nested submodel with zero amplitude and zero inverse length-scale. This leads to a prior of the form:

$$p(\eta, \kappa; \nu) = \frac{1}{2}\lambda_1\lambda_2\kappa^{-1/2}\exp(-\lambda_1\sqrt{\kappa} - \lambda_2\eta), \tag{2.10}$$

$$\lambda_1 = -\log\alpha_\rho\sqrt{\frac{\rho_0}{\sqrt{8\nu}}},$$

$$\lambda_2 = \frac{\log\alpha_\eta}{\eta_0}.$$

The parameters of this distribution, $\eta_0, \rho_0, \alpha_\eta, \alpha_\rho$, must be set by the researcher.

Luckily, they have interpretable meanings, that can be used to fix them in a weakly informative way. Specifically, under this prior, the following probability statements hold:

$$P(\eta > \eta_0) = \alpha_\eta, \tag{2.11}$$

$$P(\rho < \rho_0) = \alpha_\rho, \tag{2.12}$$

where $\rho = \sqrt{8\nu}/\kappa$. That is, this prior allows us to set prior expectations on the tail probabilities of the magnitude of heterogeneity, and the degree of intertemporal information sharing. In our work, we typically fix $\eta_0 = 10$, $\rho_0 = 1$, $\alpha_\eta = 0.01$, and $\alpha_\rho = 0.001$.

Population Evolution

While GPs are necessary at the individual-level in our dynamic heterogeneity specification, insofar as they capture consumer-level departures from a population mean function, any time series specification can be used to specify the evolution of that population trajectory, which we also refer to as the mean model. We denote a generic specification for the mean model by

$$\mu_p(t) \sim \pi_{\text{Pop.}}(\boldsymbol{\alpha}_p). \tag{2.13}$$

The mean model captures population-level dynamics, and estimating the mean model jointly with the individual-level functions is the primary source of inter-individual information sharing. The emphasis of this paper is showcasing how DHGP can be used to specify heterogeneity around a given mean model. To illustrate the flexibility of DHGP, we test four different mean models in this application, corresponding to four common and fairly general specifications

commonly employed in the literature:

1. Random walk state space (RW): dynamic linear state space models are common in the literature. The simplest class of dynamic linear state space models is the random walk specification, given by:

$$\mu_p(t) = \mu_{pt} = \mu_{pt-1} + \zeta_{pt},\ \zeta_{pt} \sim \mathcal{N}(0, \tau_p^2). \tag{2.14}$$

   While this specification is very simple, it is also quite flexible, and quick to estimate. We thus rely on it in both this application and in the subsequent application. In standard state space applications, this model may be estimated using Kalman filtering. However, when employing DHGP, we have found huge efficiency gains in estimating the mean specification jointly with the rest of the DHGP parameters, and hence estimate it using NUTS.

2. Gaussian process (GP): Similar to Dew and Ansari (2018), we can assume a GP as the population model:

$$\mu_p(t) \sim \mathcal{GP}(c_p, k_{0p}(t, t'; \eta_{0p}, \rho_{0p}, \nu_{0p})). \tag{2.15}$$

   Here we assume a constant mean c_p and a Matérn kernel, with the degree parameter ν_{0p} of this upper level kernel set to be the same as the DHGP kernel.[5]

3. Autoregressive moving average (ARMA) time series: Time series models are especially common in econometric applications, and can easily be incorporated

[5]This is merely a simplifying assumption: there is no theory-based reason to fix both to have the same smoothness.

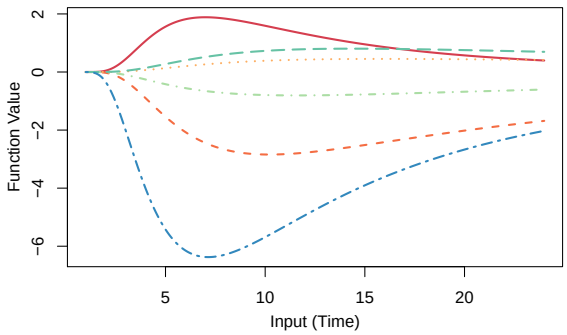

Figure 2.3: Examples of the generalized inverse Gamma distribution PDF, which is the parametric form we assume as one of our mean models. We can see the ability of this form to capture different peaks, as well as start and end levels, via the three parameters $\alpha_1, \alpha_2, \alpha_3$.

into DHGP. We test an ARMA(1) mean model specification, given by:

$$\mu_p(t) = \mu_{pt} = \alpha_{0p} + \alpha_{1p}\,\mu_{pt-1} + \alpha_{2p}\,\zeta_{pt-1} + \zeta_{pt},\ \ \zeta_{pt} \sim \mathcal{N}(0, \tau_p^2). \tag{2.16}$$

4. Parametric: A theory-driven parametric model can also serve as the mean model. In this case, one interesting question is the degree to which the Great Recession is associated with changes in consumers' preference parameters. Thus, to illustrate how a parametric model could be used in conjunction with DHGP heterogeneity, we test a generalized inverse Gamma distribution PDF, given by:

$$\mu_p(t) = \alpha_{0p} + \alpha_{1p}\left[(\alpha_{2p})^{\alpha_{3p}}\frac{1}{\Gamma(\alpha_{3p})}t^{-\alpha_{3p}-1}\exp\left(-\frac{\alpha_{2p}}{t}\right)\right] \tag{2.17}$$

with $\alpha_2, \alpha_3 > 0$. This is a parametric mean function that allows for a unimodal pattern, with different pre- and post-peak levels, as shown in Figure 2.3. This behavior is aimed at isolating the effect of the recession.

Full Model Specification

Bringing together all the model components, the class of DHGP dynamic choice models is given by the random utility specification,

$$\begin{aligned} u_{ijm} &= \mathbf{x}_{ijm}^{\top}\boldsymbol{\beta}_i(t_{im}) + \epsilon_{ijm}, \\ \boldsymbol{\beta}_i(t) &= (\beta_{i1}(t), \ldots, \beta_{iP}(t)), \\ \beta_{ip}(t) &\sim \mathcal{GP}\left(\mu_p(t), k_{\text{Mat-3/2}}(t, t'; \eta_p, \kappa_p)\right), \\ \mu_p(t) &\sim \pi_{\text{Pop.}}(\alpha_p) \\ \epsilon_{ijm} &\sim \text{EV}(0, 1). \end{aligned} \tag{2.18}$$

This leads to standard discrete choice probabilities, conditioned on the parameter values, as given in Equation 2.2. To infer the dynamic sensitivities in a data-driven way, we estimate the GP hyperparameters η_p and κ_p, and the mean function parameters $\boldsymbol{\alpha}_p$. As noted before, this leads to the double hierarchy: the model is hierarchical in the sense of learning hyperparameters in a data-driven way, and then also hierarchical in the sense that individual-level curves are pooled around a mean function.

Inference

Recall that the data consist of $m = 1, \ldots, M$ choice observations involving J alternatives, where each alternative has an attribute vector $\boldsymbol{x}_{mj}$ and an outcome denoted y_m. The collections of these across all observations are denoted by the matrix $\boldsymbol{X}$ and the vector $\mathbf{y}$, respectively. We index the customer and time period corresponding to each observation m with i_m and t_m, respectively. The model unknowns are the GP function values $\{\{\beta_{ip}(t)\}_{p=1}^{P}\}_{i=1}^{N}$, their corresponding mean

functions $\{\mu_p(t)\}_{p=1}^P$, and the model hyperparameters, $\{\phi_p, \alpha_p\}_{p=1}^P$, which are the parameters of the DHGP kernel and mean model respectively. We can then write the joint density of the data and model parameters as:

$$p(y, \beta, \mu, \alpha, \phi|X) = \prod_{m=1}^{M} p(y_m|X_m, \{\beta_{i_m p}(t_m)\}_{p=1}^P) \times \prod_{i=1}^{I}\prod_{p=1}^{P} p(\beta_{ip}(t)|\mu_p(t), \phi_p)\, p(\mu_p(t)|\alpha_p)\, p(\phi_p)\, p(\alpha_p), \tag{2.19}$$

and use this joint density to estimate our model using Hamiltonian Monte Carlo (HMC) via `Stan`, a probabilistic programming language (Carpenter et al., 2016).

HMC is an MCMC method that uses the gradient of the log-joint distribution to efficiently explore the posterior distribution (Neal, 1998; Singh, Hansen, and Gupta, 2005; Neal, 2011). In particular, `Stan` uses the No-U-Turn Sampler, or NUTS (Hoffman and Gelman, 2014b), a variant of HMC that automatically sets the stepsize and dynamically determines the number of leapfrog steps in HMC to optimize the mixing rate of the chain, which eliminates the need for costly and onerous manual tuning of the algorithm. At each iteration, the function values and the hyperparameters are updated jointly, thereby alleviating some of the inefficiencies of simpler MCMC methods that ignore the typical strong dependence between these two sets of unknown quantities (Neal, 1998; Flaxman et al., 2016). We have found jointly sampling all parameters via NUTS is much more efficient compared to competing strategies, including custom built HMC-within-Gibbs methods, and adaptive metropolis methods, even when leveraging parallelization across individuals as is possible with those competing methods. We run the sampler for 400 iterations (200 warmup), and measure convergence through the $\hat{R}$ statistic (Gelman and Rubin, 1992). In all cases, we achieve $\hat{R} \approx 1$.

Links with Previously Proposed Models

The DHGP specification of dynamic heterogeneity is very flexible, and can be seen as a generalization of both the FO and independent periods specifications described above. To see the relationship between the DHGP and the FO model, consider the individual-level kernel, where for simplicity we omit the coefficient subscript p:

$$k(t, t'; \eta, \kappa) = \eta \, (1 + \kappa \, |t - t'|) \exp\{-\kappa \, |t - t'|\} \, . \tag{2.20}$$

From this expression, it is immediately obvious that as $\kappa \to 0$, the kernel degenerates to $k(t, t'; \eta) = \eta$. This results in a rank one covariance matrix for the individual-level curves, and is equivalent to the bias kernel we described previously, which is the same as a model with a constant offset. In other words, as $\kappa \to 0$, the model converges to the FO model.[6] We demonstrate this convergence in Figure 2.2. This also explains why we use the inverse length-scale parametrization, as this allows us to place a sizable prior mass on models converging to the fixed-offsets model, and thus allows us to add a prior tendency toward that restricted model. Therefore, if the posterior places a sizable mass away from zero, we can be confident that the data rejects the FO restriction.

In addition, under certain mean models, DHGP nests the independent periods model. As the the inverse length-scale of the individual-level model $\kappa \to \infty$, individuals' preferences become uncorrelated over time (i.e. $k(t, t') \to 0$ for $t \neq t'$). When paired with a mean model that permits the same convergence, like a GP with a length-scale approaching zero, or a random walk model with high transition variance, the DHGP model is equivalent to using an independent normal

[6] While we described the relationship in terms of the Matérn kernel with degree 3/2, this relationship holds for any member of the Matérn family of kernels, and in fact for a variety of other common kernels that could be employed as alternatives to this particular choice.

heterogeneity distribution each period. Note that while we do not place prior mass toward infinity in the individual-level model, any value of the inverse length-scale $\kappa \gg 1$ will lead to a negligible off-diagonal covariance matrix, and hence practically, can achieve a similar result. This lack of information sharing across periods leads to individual-level curves that are essentially random around the mean function, as shown in the first panel of Figure 2.2.

2.3 Simulation Studies

In this section, we use simulated data to illustrate why it is important to capture dynamic heterogeneity, and the pitfalls of restricting intertemporal patterns at the individual-level by imposing static heterogeneity on a dynamic model. Specifically, we explore in what situations static heterogeneity, as estimated through a fixed offsets approach, fails to accurately capture meaningful dynamic heterogeneity when it does exist, and the implications of such failures for understanding market trends and developing marketing strategy. Throughout our simulation studies, we exclusively use a GP population model, for both the DHGP and the FO specifications.

We compare the properties of the DHGP to the independent periods (IP) and fixed offsets (FO) specifications. In the IP approach, no or very minimal information is shared across time periods, leading to a model that is fully flexible in capturing temporal patterns, but is highly inefficient. This problem is exacerbated for smaller sample sizes, as there is no sharing of information across time periods to alleviate the per period data sparsity. While the IP specification is inefficient, it does not result in misleading estimates in the presence of dynamic heterogeneity. However, models like the FO that restrict the patterns of individual-level variation

over time can yield both misleading customer-level inferences, and biased population-level estimates. Because the restricted patterns of heterogeneity are built into the model, no amount of data can correct these biases. We now explore these pathologies in more detail.

Individual-level Effects We start by exploring how each model recovers customer-level effects. We simulated data from a small sample of 10 consumers according to our DHGP model. Each customer makes 5 choices per period for 20 periods from a choice set of 3 brands. We deliberately use such a small sample in this section to make the differences among the models clearly evident, but similar patterns of results hold even when the sample size is raised to 100, 200, or even 500 consumers.[7]

In Figure 2.4, we plot three examples of individual price sensitivities, relative to a population sensitivity, that were randomly generated from the DHGP model. In the first panel, we see a consumer whose price sensitivity roughly mirrors that of the population, deviating slightly in later time periods. We expect the fixed-offsets model to capture this individual's trajectory reasonably well, as it does not significantly deviate from the population trend. However, the FO model cannot, by definition, capture the upward swing toward the end, which we call a divergent trajectory. This late-stage deviation indicates that this customer is becoming less price sensitive over time, relative to the population, which in turn means this divergence could be important for targeting and forecasting purposes. Likewise, an individual who exhibits the opposite trend—converging toward the population mean—could also be meaningful, though we do not plot an example of that.

[7]The results from all simulations are available from the authors upon request, and the code will be made available online.

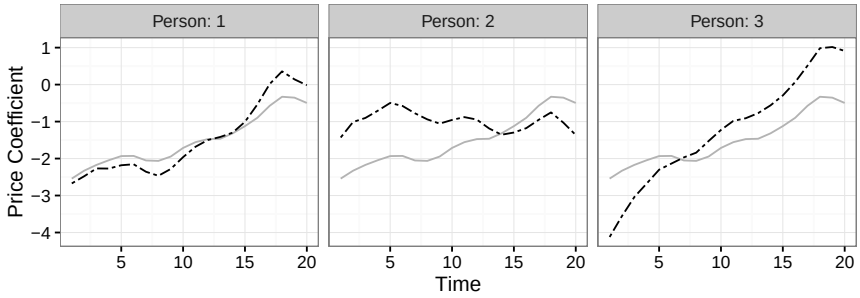

Figure 2.4: Three simulated individual-level price sensitivities denoted in black dot-dash; population-level dynamics denoted in grey solid. In this example, the population as a whole is becoming less price sensitive over time. From left to right, these illustrate a case that would be captured reasonably well by the FO specification besides a late period deviation (left panel), and two examples of crossover effects that would not be captured by the FO model (middle and right panels).

In the second and third panels, we have two examples of crossover customer-level trajectories. Customer 2's price sensitivity has remained relatively constant, relative to the population which has progressively become less price sensitive. This indicates that, although Customer 2 used to be relatively price insensitive, he is now relatively price sensitive. Individual 3 exhibits the opposite: although she was previously quite price sensitive, relative to the population, she has become, over time, less price sensitive, at a faster rate than the population as a whole. Clearly these, too, are important effects: in understanding how these consumers may respond to a price change in the current period, these trajectories must be understood.

In Figure 2.5, we show how DHGP, as well as the two benchmarks, capture these effects. The first thing to note is that DHGP recovers all of the effects quite well. Each of the effects described above would be evident in the posterior customer-level curve estimates. While the IP model does capture the general trajectories, it is very noisy and jagged, and accompanied by a large amount of posterior uncertainty, which would be further exacerbated if we relax the assumption of a common variance across time periods. Finally, in the FO model, the individual-level curves exactly mirror the population curve, which clearly does

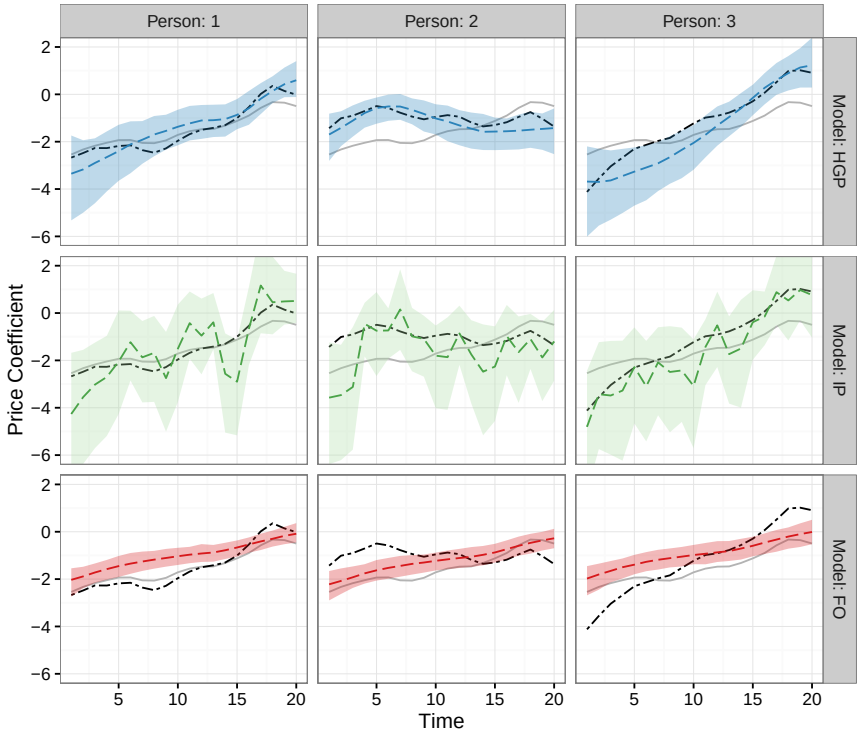

Figure 2.5: Posterior medians of the three models, on the three case studies displayed in Figure 2.4.

not capture the effects of interest. In fact, under FO, all three individuals are estimated to have nearly identical curves, despite the dramatically different true trajectories. Further emphasizing this point, the posterior medians of their fixed-offsets are given by 0.01, -0.2, and 0.06 respectively. Using these estimates as indicators for how different the individuals' price sensitivities are from the population, relative to the estimated standard deviation of 0.82 for the population heterogeneity distribution, we would infer these individuals' are average in price sensitivity. While this may be true for Customer 1, this is clearly inaccurate for Customers 2 and 3: these individuals are exceptional, representing trends that, if uncovered, may impact managerial decision making.

Biased Population Estimates In static choice models, it is well established that not accounting for heterogeneity in the coefficients can result in an attenuation bias in

the estimates of the population mean trajectories, wherein the parameters are biased toward zero. The bias worsens when the magnitude of heterogeneity grows larger. We show similar results in simulations for dynamic heterogeneity: if a restricted form of heterogeneity is used, as in the fixed-offsets model, when there is significant dynamic heterogeneity, the model will fail to recover both the true magnitude of heterogeneity and the true population trend, resulting in population parameter estimates that are biased toward zero. To illustrate this attenuation bias, we again simulated data according to the DHGP, and then fit the fixed-offsets model. For the simulations in this section, we simulated 200 consumers, each making 5 choices per period over 20 periods from a choice set of 3 brands with a single attribute (e.g. price). We use a higher consumer count to ensure that sample size is not the driving factor in the inaccurate recovery of the population trend.

In Figure 2.6, we show an example of the bias. In this simulation, we sampled from the DHGP with a relatively high amplitude in the individual-level GP model, $\eta = 4$. This means that individual-level curves can deviate quite far from the population curve. We note that, while this value is relatively high, it is consistent with some of our estimates in the applications that follow. We see that when the FO model is used to estimate choice data with individual-level dynamics, it estimates a population effect that is biased toward zero. Figure 2.6 shows this effect with the simulated price sensitivity, but such a bias is present across all of the coefficients. Moreover, the spread of individual-level effects is underestimated. In Figure 2.7, we show at an evenly-spaced set of time periods the distribution of individual-level effects around the population function, together with the period-specific distribution recovered by the DHGP, and the distribution recovered by the FO model.[8] We see that, while the DHGP clearly reflects an appropriate

[8]Since the FO model does not allow individual-level effects to change over time, this distribution is the same, except for a shift, reflecting the changing population median.

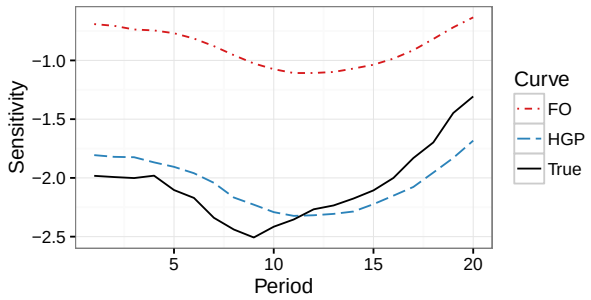

Figure 2.6: Simulation with moderate heterogeneity; the fixed offset specification's population dynamics estimate is biased toward zero. (A color version of this figure is available online.)

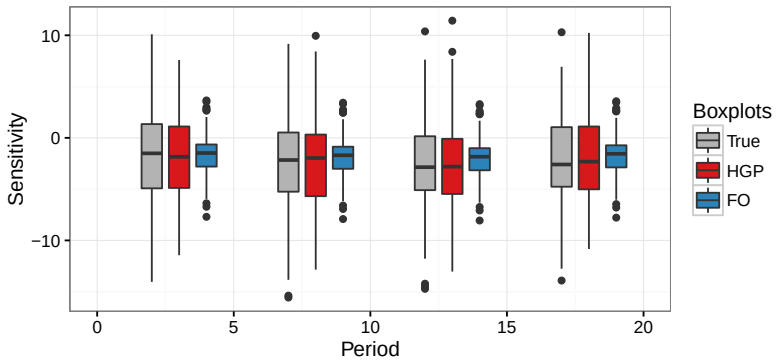

Figure 2.7: Boxplots showing the distribution of the individual-level effects in specific periods, evaluated at $t = 3, 8, 13, 18$, illustrating the underestimation of spread by the FO restriction. (A color version of this figure is available online.)

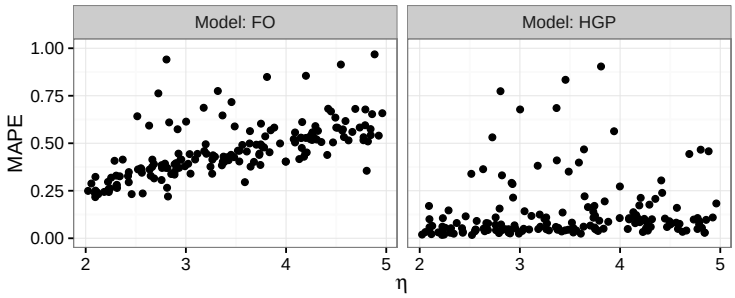

Figure 2.8: MAPE of the recovery of the population mean dynamics across the two models for the "price" coefficient; note the truncation at MAPE of 1 omits 16 observations out of 172 simulations from the plot.

amount of spread in its individual-level estimates, the FO model dramatically underestimates the magnitude of heterogeneity.

While Figures 2.6 and 2.7 illustrate an example of the bias, we wanted to ensure that this bias is consistently present, and caused by the presence and

magnitude of dynamic heterogeneity. To explore this, we generated data from the DHGP several times, drawing the lower-level amplitude parameter, η, from a uniform distribution on $[2, 7]$, effectively varying it continuously over that range using 172 simulations. As η grows larger, customer-level trajectories depart more from the population curve. After estimating the two models, FO and DHGP, we then calculated the mean absolute percentage error (MAPE) of each in recovering the true population curves across all of the coefficients.[9] We plot the results in Figure 2.8. We see clearly that as the magnitude of dynamic heterogeneity grows, the FO model does worse at recovering the true population mean dynamics. The DHGP, on the other hand, does not suffer from this bias, and can recover the population dynamics at all levels of η.

We also ran several robustness checks: the same biases are evident no matter what prior distribution is used on the parameters of interest. Furthermore, when the true data generating process is the fixed-offsets model, the population curve is successfully recovered by both the FO model and our DHGP. Finally, as the data generating process converges to the FO model, i.e., when the inverse length-scale parameter $\kappa \to 0$, the FO model indeed improves progressively at recovering the true mean curve.

Summary of Simulations Across the many simulations, we have shown three important effects: first, the DHGP incorporates information sharing both across consumers and over time periods, which allows it to produce much smoother customer-level estimates than those obtained from IP models that do not pool across time. Second, restricted models of heterogeneity can provide misleading

[9]We use MAPE because it is a scale free measure of accuracy. We do note, however, that since MAPE involves division by a "true value", it can exhibit high variance when this true value is close to zero. This leads to some aberrant simulations, and so we focus in this case on the best performing 90% of the simulations (across both models).

managerial insights when applied to data with individual-level dynamics. Finally, in the presence of customer-level dynamics, a model that flexibly accounts for these dynamics is necessary, even in estimating dynamics at the population-level. If a restricted specification like FO is employed, estimates of both the population curve and the magnitude of heterogeneity will suffer from an attenuation bias.

2.4 Application

In this application, we apply DHGP to understand changes in consumer preferences in the context of grocery store purchasing. We focus on a time window that encompasses the Great Recession, a period of time where we expect non-trivial preference dynamics. Throughout, we compare the insights gained from a dynamic heterogeneity specification, estimated using DHGP, to a static heterogeneity specification, estimated using FO.

2.4.1 Data

We model brand choice in the IRI consumer packaged good (CPG) panel data, from January 1st, 2006 to December 31st, 2010 (Bronnenberg et al., 2008). This span includes the Great Recession, which according to NBER, began in December 2007, and ended in June 2009, and thus has the potential to yield dynamics of interest to both economists and managers. However, our focus will be on capturing and characterizing the dynamic heterogeneity that exists over this window. Specifically, we study the evolution of consumers' individual-level brand preferences, price sensitivities, and feature/display sensitivities across six different categories: peanut butter, coffee, potato chips, laundry detergent, tissues, and toilet paper. We

aggregate the data monthly, saving the last four months of data for holdout validation. We retain all panelists who spent at least five times during the data.

2.4.2 Case Study: Preferences for Tissues

In this section, we will focus our analysis on just one category and one model: the tissues category, with an AMRA mean model, and DHGP dynamic heterogeneity. We will use this specific example to build intuition as to the output of our DHGP choice model, and the insights about dynamic heterogeneity that can be generated from a DHGP specification. We defer discussion of the results across all categories and specifications to the next section. We choose this specific example because there are very interesting patterns of dynamic heterogeneity at work in the tissues category. We use the ARMA mean model because, as we describe in the next section, the ARMA mean model tended to perform the best of all the mean models studied.

The first output of the model we will consider is the estimated mean model; that is, the posterior estimates of $\mu_p(t)$. The mean model estimates for tissues under the ARMA specification are shown in Figure 2.9. What we see from these five panels is that there are obvious monthly dynamics at work. Interestingly, on average, brands 2 and 3 tended to move opposite one another, while brand 4 appears to track brand 2 to some degree. Price sensitivity appears to have experienced some monthly dips and spikes, while feature/display sensitivity appears to have been more static.

While there are certainly interesting mean patterns at work, the primary focus of this paper is capturing how individuals changed *relative* to those mean trends. In Figure 2.10, we plot, at top, a sample of individual-level curves, overlaid

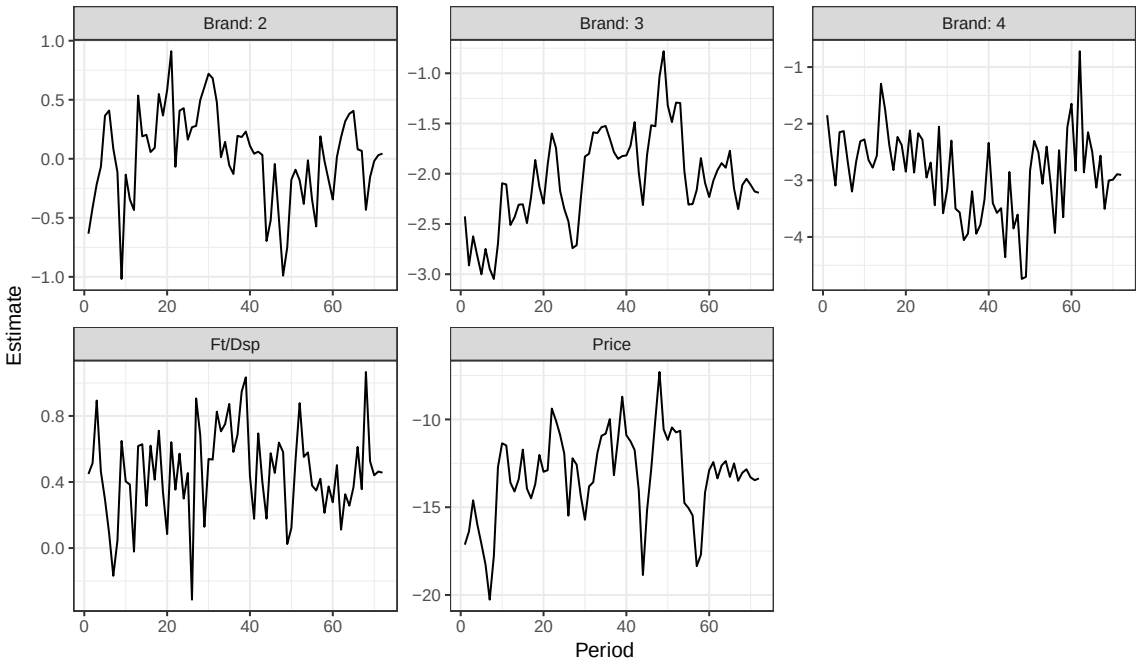

Figure 2.9: Posterior mean estimates of $\mu_p(t)$ in the tissues category under the ARMA mean model specification using DHGP heterogeneity. We see clear evidence of monthly dynamics in most of the parameters. The last four periods (months) are forecasts.

on the estimated mean model. We also plot, at bottom, the difference between those same estimated individual-level curve and the estimated mean model,

$$\text{Diff}_{ip} = \hat{\beta}_{ip}(t) - \mu_p(t). \tag{2.21}$$

We see that, while some individuals tended to remain fairly static over time, others have moved relative to the mean function. Capturing this movement is the goal of the DHGP specification. Note that these individuals were randomly sampled from the group of individuals who spent consistently throughout the sample. The reason we select only from frequent purchasers is because DHGP exhibits mean reversion: absent new observations, the DHGP estimated curves will revert to their mean, at a rate inversely proportional to κ. Hence, to be sure what we are plotting is true dynamics and not mean reversion, we must ensure that the consumers spent consistently.

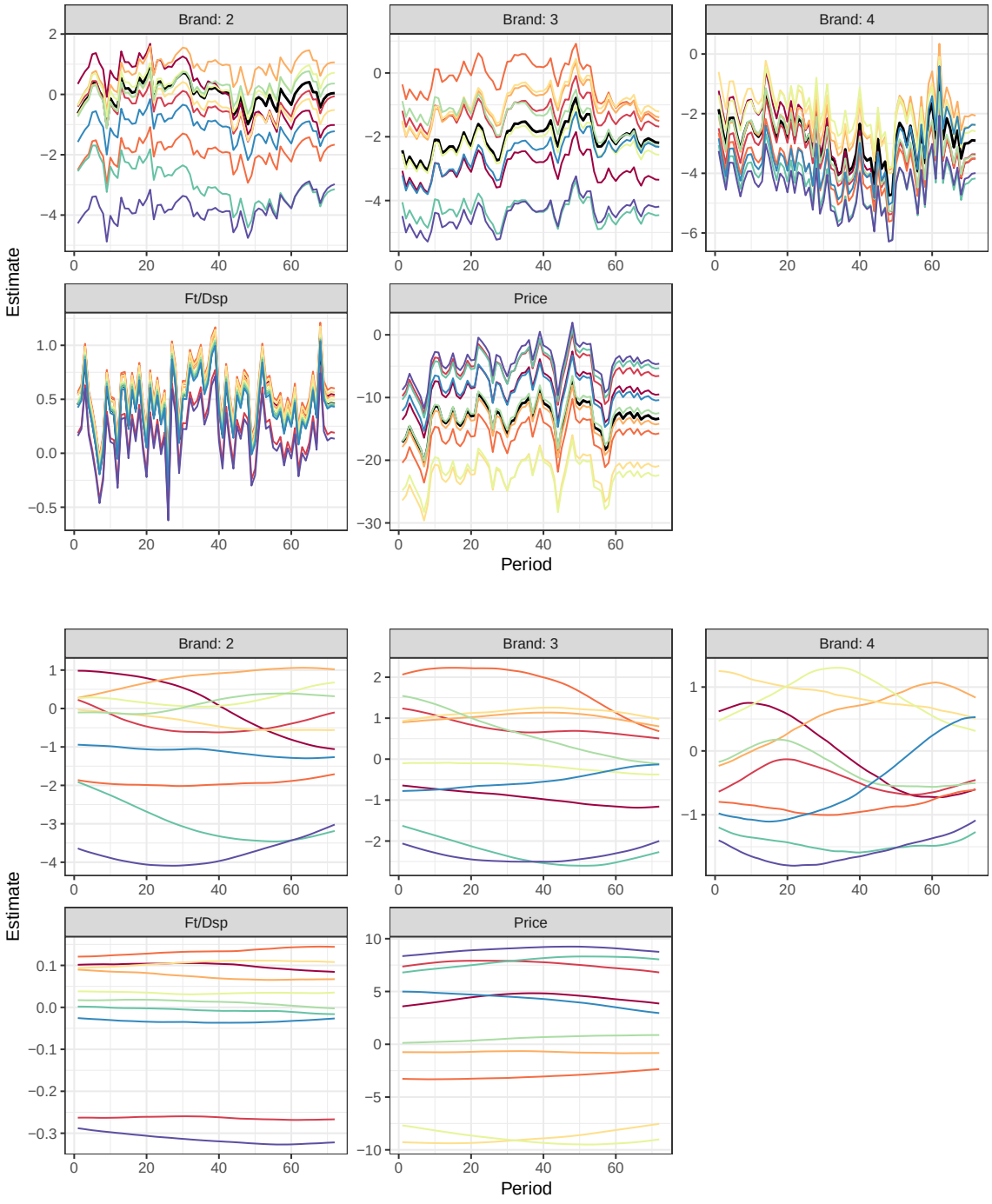

Figure 2.10: At top, we plot a random sample of individual-level curves in the tissues category, overlaid on the estimated mean model. At bottom, we plot the difference between those same individual-level curves and the mean model.

Parameter	η_p	κ_p
Brand 2	2.011	0.026
Brand 3	1.513	0.031
Brand 4	1.431	0.068
Price	6.943	0.021
Ft/Dsp	0.298	0.009

Table 2.1: Posterior mean estimates of the DHGP hyperparameters in tissues.

The nature of the individual-level deviations is determined by the estimated hyperparameters, η_p and κ_p. As η_p grows, the individual-level curves are allowed to spread further from the mean. As κ_p grows, the individual-level curves become less smooth. The posterior mean DHGP hyperparameters for tissues are given in Table 2.1. We can see from this that feature/display has both the least degree of heterogeneity, due to its low $\eta = 0.298$, and bears the closest resemblance to the fixed offsets assumption, with $\kappa = 0.009 \approx 0$. The price coefficient has the largest degree of heterogeneity, with $\eta = 6.943$, although again the deviations from the mean are relatively smooth, with $\kappa = 0.021$. Brand 4 exhibits the least smooth variation, with the highest $\kappa = 0.068$. All of these effects are clearly evident by looking at the differences curves in Figure 2.10.

Figure 2.10 illustrates dynamic heterogeneity for a randomly sampled group of people. Now, we want to zero in on a few interesting cases, that highlight the nuanced insights possible by considering dynamic heterogeneity. To do that, in Figure 2.11, we narrow our focus to just a single parameter: the Brand 2 intercept. Then, we isolate individuals whose curves exhibit interesting behaviors:

- Converging: In the leftmost panel, we plot a set of individual curves that converge toward the population mean. These individuals started in one extreme of the distribution for brand 2 brand equity, but by the end of the observation window, were in the middle of the distribution. Under a fixed

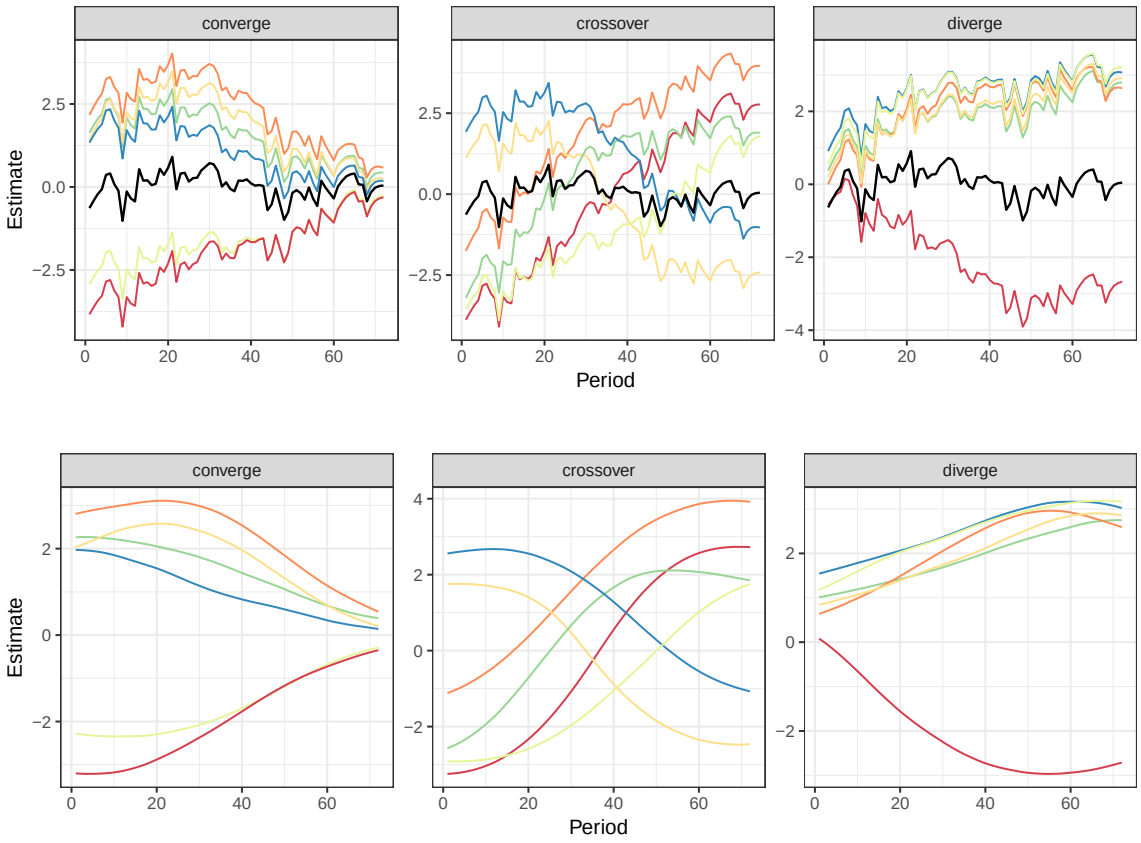

Figure 2.11: At top, we plot a sample of interesting individual-level curves in the tissues category, overlaid on the estimated mean model. Specifically, we isolate individuals whose curves converge toward the population mean, cross over the population mean, and diverge from the population mean. At bottom, we plot the difference between those same individual-level curves and the mean model, more clearly illustrating these changes.

offsets model, these individuals would be estimated as being moderately above or below the population mean, which is true only in the middle of the observation window, and does not reflect current or expected future behavior.

- Crossover: In the middle panel, we plot a set of individual curves that cross over the population mean. That is, these individuals started out relatively liking/disliking brand two (relative to others), and moved to disliking/liking (respectively) by the end of the observation. Under a fixed offsets model, these individuals would be classified as falling near the population mean; in fact, they are perhaps the least average consumers, from a marketing research perspective, as they reflect a strong change in preferences.

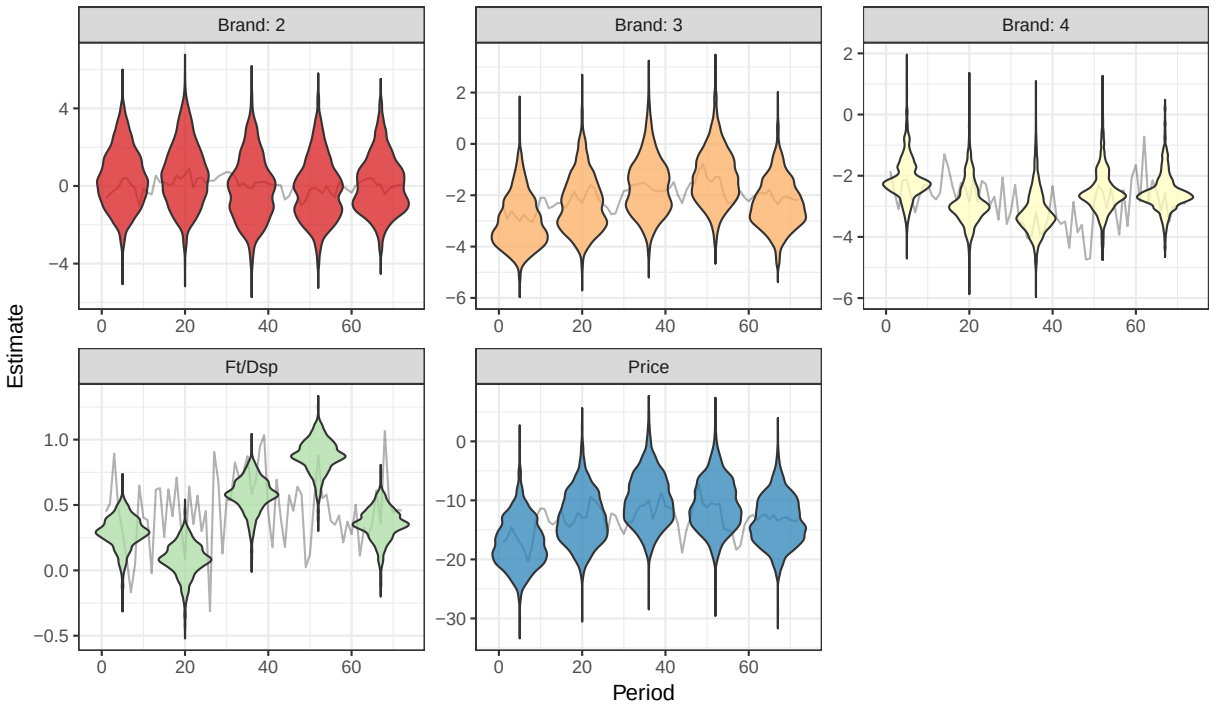

Figure 2.12: We plot the marginal distribution of individual-level effect estimates at five equally spaced time periods, using the estimates $\hat{\beta}_i p(t)$ from the DHGP specification. We see the empirical marginal distribution evolves over time.

- Diverging: Similar to the converging case, in the rightmost panel, we plot a set of individual curves that diverge away from the population mean. These individuals started out relatively average in their tastes for brand 2, but moved to the extremes of the distribution over time. Under a fixed offsets model, they would be estimated as being moderately above or below the population mean, which is only true in the middle of the observation window, and again does not reflect current or expected future behavior.

Finally, while DHGP assumes in the prior that, within a time period, the marginal distribution of individual-level parameters is normally distributed, in practice, we may find the intra-individual smoothing leads to non-normal marginals. Moreover, we may find that the marginal distribution itself evolves over time. We show the evolving empirical distribution using a series of density (violin) plots in

Figure 2.12. Capturing such evolution is only possible if we use a dynamic heterogeneity specification.

In this section, we described the key parameter outputs of the DHGP specification, and built some intuition as to what the DHGP heterogeneity specification can tell us about dynamic heterogeneity. In the next section, we give an overview of DHGP across all of the categories in our data, and discuss decision-relevant implications of capturing dynamic heterogeneity via DHGP.

2.4.3 Results Across Categories

Model Fit

The key result across the six categories is that dynamic heterogeneity is pervasive. Comparing specifications with static heterogeneity versus dynamic heterogeneity, DHGP heterogeneity fits the data better across all metrics, both in the calibration data, and in forecasting tasks, including in metrics that penalize for model complexity. We plot in-sample and forecast hit rates across all models, categories, and heterogeneity specifications in Figure 2.13. We include other fit statistics in the appendix, including WAIC, precision, sensitivity, and specificity. The superior fit of DHGP across all of these metrics strongly supports our claim that dynamic heterogeneity is present, even in relatively simple panel datasets like grocery store purchases.

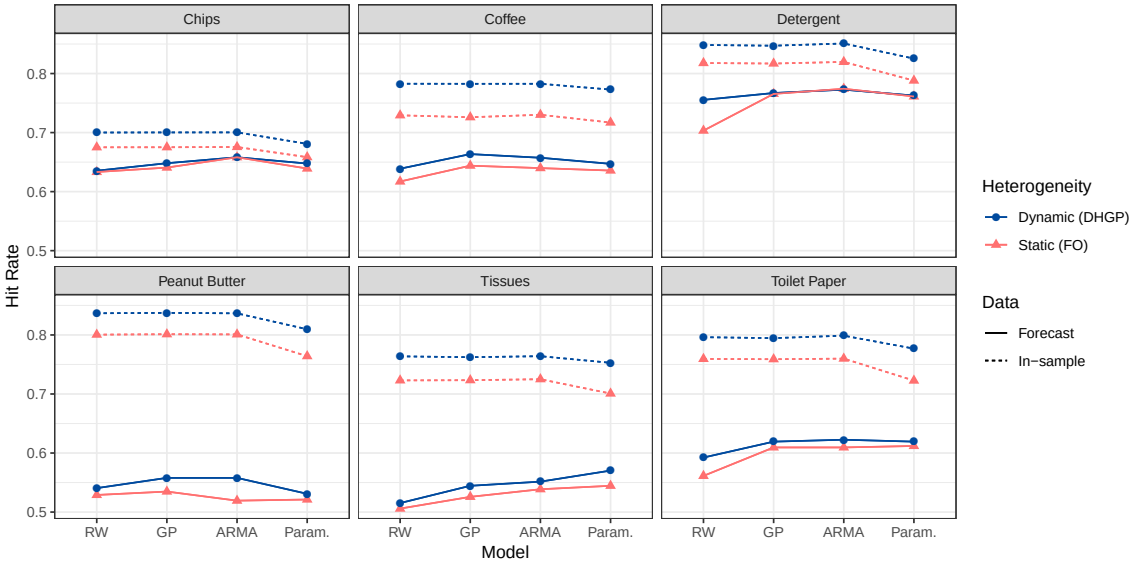

Figure 2.13: We compare in-sample and forecast hit rates in all categories (panels), for our four population mean specifications (x-axis), both in-sample (dotted lines) and forecasting ahead four months (solid lines), across static (i.e. fixed offsets; red triangles) and dynamic (DHGP; blue circles) heterogeneity specifications. We see that DHGP dynamic heterogeneity fits and predicts better than static, fixed offsets heterogeneity in all cases.

Parameter Estimates and Attenuation Bias

The hyperparameters of DHGP capture both the magnitude of dynamic heterogeneity for a given parameter, and how much intra-individual variation there is. They also allow us to assess the degree by which individual-level trajectories differ from the fixed offsets restriction, which when combined with the magnitude of heterogeneity, lets us predict how biased mean parameter estimates will be under the fixed offsets assumption.

In Figure 2.14, we show the distribution of the posterior means of both DHGP hyperparameters across categories. The magnitude of dynamic heterogeneity, η, is typically large, especially for brand intercepts, and, in some cases, for price sensitivity.[10] Moreover, DHGP soundly rejects the fixed offsets

[10]It is difficult to directly compare η across coefficients, as it is not invariant to the scaling of the predictors: brand intercepts are binary whereas the other features are standardized (mean zero, variance one).

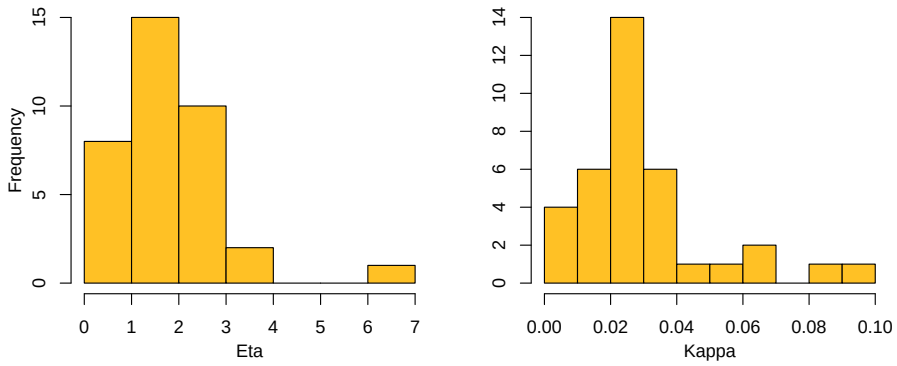

Figure 2.14: Histogram of the mean posterior hyperparameter values across all categories and parameters.

Category	Price	FtDsp	Brand 2	Brand 3	Brand 4	Brand 5	Brand 6
Chips	0.03	0.01	0.04	0.21	0.02		
Coffee	0.15	0.04	0.17	-0.02	0.40	0.11	
Peanut Butter	0.26	0.03	0.04	0.33	0.15	0.09	
Detergent	0.47	0.05	0.46	0.18	0.28	0.33	0.17
Tissues	0.77	0.03	0.04	0.15	0.31		
Toilet Paper	0.13	0.05	0.02	0.06	0.04	0.10	0.15

Table 2.2: Values of the signed difference statistic across categories and parameters. As can be seen, all but one are positive, which gives empirical support that a restrictive static heterogeneity specification around a dynamic mean model leads to an attenuation bias even in mean parameter estimates.

model: κ, the inverse length-scale, is centered away from zero, with a mode of around $\kappa = 0.025$, but with some values as high as $\kappa = 0.09$. Figure 2.2 provides some intuition as to what specific values of κ imply about individual-level variation around the population curve.

In our discussion of dynamic heterogeneity, we described how not accounting for dynamic heterogeneity in models of parametric evolution can lead to an attenuation bias in both the mean model, and in the magnitude of heterogeneity. We give simulation evidence supporting that claim in the appendix. However, we also find evidence of the bias empirically. Specifically, we find that the empirical standard deviation of individual-level parameters within a given time period is, on average, estimated to be lower using a fixed offsets model than with DHGP, as

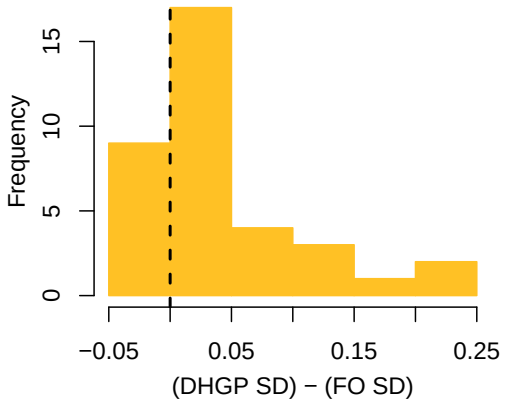

Figure 2.15: The histogram of the differences between the average within time period SD of the individual-level estimates calculated using DHGP and FO, where each data point is a category/parameter pair.

shown in Figure 2.15. Nearly all are above zero, indicating a downward bias in the spread of FO estimates versus DHGP.

Moreover, if we look at the mean curves recovered from a DHGP heterogeneity specification versus a FO specification, we see the FO mean curves are biased toward zero. To illustrate that, we consider the following statistic, which we refer to as the *signed relative difference* statistic:

$$\text{SRD}_p = \frac{1}{T}\sum_{t=1}^{T} \text{sign}\big(\hat{\mu}_p^D(t)\big) \times \frac{\hat{\mu}_p^D(t) - \hat{\mu}_p^S(t)}{1 + |\hat{\mu}_p^D(t)|}, \tag{2.22}$$

where $\hat{\mu}_p^D(t)$ is the estimated value of the mean model at time t under a dynamic heterogeneity specification (i.e. DHGP), $\hat{\mu}_p^S(t)$ estimated value of the mean model at time t under a static heterogeneity assumption (i.e. FO), and $\text{sign}(x) = 1$ if $x \geq 0$ and -1 if $x < 0$. This statistic will always be positive when $\hat{\mu}_p^D(t)$ is farther away from zero than $\hat{\mu}_p^S(t)$. Moreover, its magnitude reflects how much further $\hat{\mu}_p^D(t)$ is away from zero than $\hat{\mu}_p^S(t)$, on average, on a relative basis.

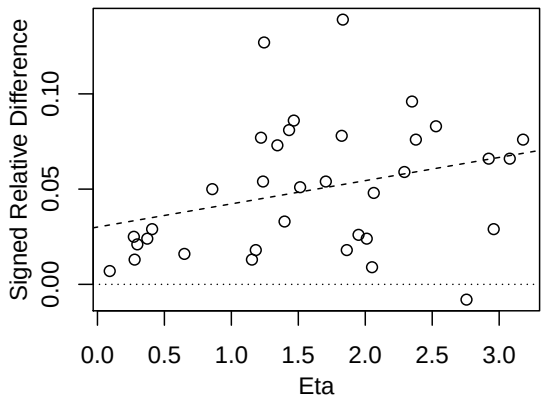

Figure 2.16: The signed relative difference statistic as a function of the posterior mean estimate of the hyperparameter η. We expect the attenuation bias to grow as the magnitude of dynamic heterogeneity grows. We omit one outlier with $\eta > 7$ to aid visualization.

In Figure 2.16, we plot the signed relative difference as a function of η, the magnitude of dynamic heterogeneity. First, we see that all but one of the SRD statistics is above zero, lending strong empirical support to the existence of the attenuation bias. We argued previously that, as the magnitude of dynamic heterogeneity grows, the attenuation bias worsens. We also see in Figure 2.16 a slight upward trend, consistent with this prediction.

Individual-level Elasticities

Accounting for dynamic heterogeneity via DHGP is important for accurately computing decision-relevant quantities, including time-varying price elasticities. By both correcting for the attenuation bias, and estimating intra-individual dynamics, the individual-level decision variables inferred from DHGP may be dramatically different compared to a static heterogeneity specification. To illustrate that, we consider own price elasticity of demand across static and dynamic heterogeneity specifications. For each individual, and each time period in which that individual purchased, we compute the following elasticity, following the standard logit

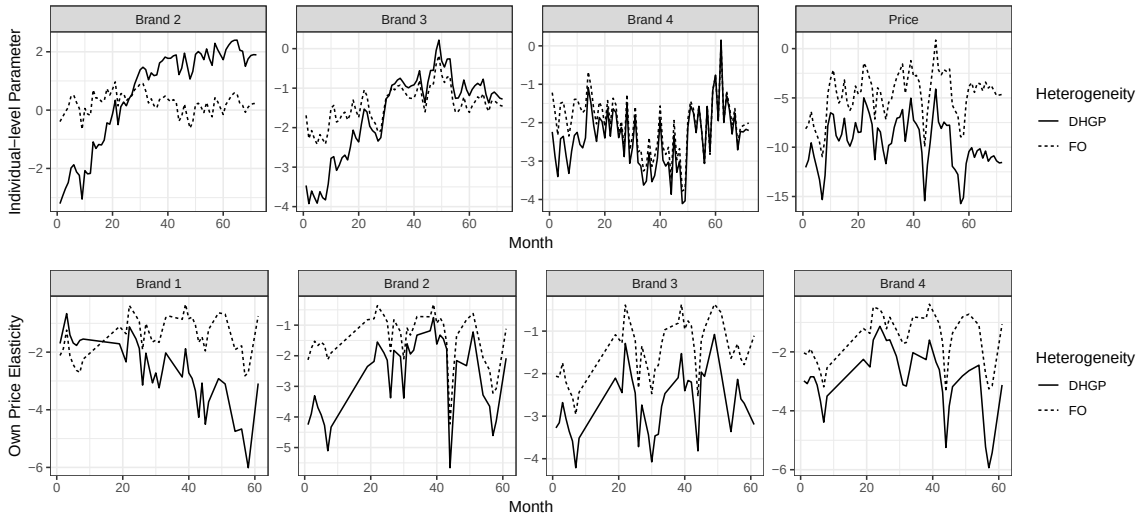

Figure 2.17: Top: The individual-level parameters for an illustrative consumer for the three brand intercept and price coefficient in the tissues category, across the two heterogeneity specifications. Bottom: the implied own price elasticities for the same consumer, computed using Equation 2.23.

elasticity formula (Train, 2009):

$$\epsilon_{ib}(t) = \hat{\beta}_i^{\mathrm{P}}(t) \times \mathrm{Price}_{it} \times [1 - p_{ib}(t)]\,, \tag{2.23}$$

where $\hat{\beta}_i^{\mathrm{P}}(t)$ is the model estimate for person i's price parameter at time t, in this case estimated as the posterior mean, and $p_{ib}(t)$ is the probability that person i chooses brand b at time t under the model. If a given individual has multiple observations per time period, we average them together. We compute this quantity for both of the models.

First, we consider an illustrative case of a tissues consumer. In Figure 2.17, we present two sets of plots: in the first, we show the same consumer's choice parameters under both dynamic (DHGP) and static (FO) heterogeneity assumptions. In the second, we show the implied elasticities over time, computed using Equation 2.23, for all periods in which the consumer was active. Comparing DHGP to FO heterogeneity in the top panel, we see two things: first, the consumer's brand intercepts deviated significantly from the pattern implied by FO,

Category	Mean	SD	5%	25%	50%	75%	95%
Chips	3.08	33.55	-36.00	-5.32	3.20	10.85	46.76
Coffee	11.65	61.33	-146.56	-6.24	15.00	30.74	149.25
Peanut Butter	11.52	30.14	-23.74	2.95	10.06	17.76	54.64
Detergent	12.05	33.98	-33.63	3.76	12.86	19.88	57.63
Tissues	6.28	25.62	-22.08	-1.29	5.65	12.47	37.62
Toilet Paper	9.27	31.45	-26.57	0.09	7.72	16.09	57.61

Table 2.3: Summary statistics for the distribution across people and parameters for the percentage difference in individual-level elasticity estimates, averaged over time, between static and dynamic heterogeneity specifications.

due to individual-level dynamics. This effect is especially interesting for brand 2, where the consumer went from negative to positive. Second, we see that the price curve is *significantly* underestimated using FO, which is likely driven by the attenuation bias. Taken together, these effects produce two effects in the elasticities: first, in almost all cases, the price elasticity is underestimated by roughly 50%. Second, we see the brand intercept dynamics spill over into the price elasticities, with very different patterns implied especially for brands 1 and 2.

This example builds intuition around why we expect to see differences between decision variables under dynamic versus static heterogeneity assumptions. Such differences in elasticities are not limited to special cases. In fact, they are widespread across all categories. To assess these differences more generally, we compute the difference $\epsilon^D_{ib}(t) - \epsilon^S_{ib}(t)$ for all individuals, for all time periods in which those individuals spent, for all coefficients. We present summary statistics for the distribution of this difference by categories, across all parameters, in Table 2.3. We can see that, on average, individual-level elasticities are underestimated by using static versus dynamic heterogeneity specifications. Moreover, the tails on the distribution are huge, indicating that, for some people, the difference in estimated price elasticitiy between static and dynamic heterogeneity specifications is massive.

Het. Type	Chips	Coffee	Detergent	Peanut Butter	Tissues	Toilet Paper
Dynamic	2,203.12	1,639.18	192.57	314.06	235.59	199.52
Static	2,176.69	1,545.88	173.29	302.93	208.28	186.39

Table 2.4: Optimal category-level profits under dynamic versus static heterogeneity models, assuming an ARMA mean model.

Targeted Pricing

In this section, we illustrate the potential impact of accounting for dynamic heterogeneity for retailer profitability through an application to targeted pricing. Since the work of Rossi et al. (1996), marketers have developed practical ways of using the distribution of consumer preferences to implement targeted actions such as optimal discounts (Chintagunta et al., 2005; Duvvuri et al., 2007). Our findings suggest that ignoring dynamic heterogeneity may yield biased or misleading elasticity estimates and hence adversely impact targeting decisions. In this application, we assume the role of a category pricing manager for each of the categories examined, and compute optimal targeted profits assuming both static and dynamic heterogeneity.

Specifically, we use the model parameters calibrated on the in-sample data, and compute the optimal discounts and resulting profits for the last four heldout months. As we lack of marginal cost data such as wholesale prices, we assume a regular 25% gross margin as in Duvvuri et al. (2007) for all brands in these categories. This experiment also takes choice occasion as exogenous, because the modeling efforts focus on brand choice rather than purchase incidence (Chintagunta et al. 2005). In computing optimal discounts, we search over a grid of price reductions ranging from 0% to 25% in steps of 1%, for each brand in every choice occasion, to determine the optimal discount that yields the highest profit.

The results of this exercise are in Table 2.4. As we can see, dynamic heterogeneity improves profits in all categories. The reasons for this are intuitive: dynamic heterogeneity captures the movements in price elasticities, as described previously, and more richly characterizes, and then predicts future variations in consumer preferences. This added richness provides more nuance for computing targeted discounts.

2.4.4 The Great Recession

Finally, we consider how preferences appear to have changed during the Great Recession, with a focus on the individual-level evolution of preferences. While not our primary focus, our data span the period of the Great Recession, as well as periods both before and after. We can thus use our DHGP estimates to describe how preferences appear to have changed during that period, as a final illustration of the insights gained by modeling dynamic heterogeneity, and how such a specification is useful for economic research.

Elasticities

Prior literature has documented how price sensitivity within categories varies with business cycles (Gordon et al., 2013), and more generally how preferences for CPG shifted, on average, during the Great Recession (Cha et al., 2015). Similarly, in this work, we can use the individual-level estimates from DHGP heterogeneity to compute how the average price elasticity of demand changed over time across categories during the recession, for our six focal categories.

Across categories, we find mixed effects of the recession on average own price

elasticities. In the detergent category, for instance, we find many brands experienced significant drops in average price elasticity during the recession, as plotted in Figure 2.18. Note that we retain the sign on price elasticity here: hence, a decrease in price elasticity means consumers are increasingly substituting away from the focal brand with the same percentage increase in price. Similar to detergent, many brands in chips and toilet paper also exhibited dips or slumps in price elasticity during the recession.[11] A decrease in price elasticity during the recession is the intuitive direction, as the Great Recession negatively affected many people's earnings, which intuitively should lead to higher price sensitivity. Peanut butter and coffee, on the other hand, do not appear to have been significantly impacted: while there are some apparent dynamics in average price elasticity during the recession era, they are not clearly differentiated from the dynamics before and after. Finally, and perhaps most interestingly, tissues appear to have behaved almost countercyclically during the recession: for all brands, recession era price elasticity was greater than either pre- or post-recession price elasticity, as shown in Figure 2.19.[12]

Dynamic Heterogeneity in the Recession

Novel to DHGP, we can also summarize the nature of dynamic heterogeneity during the observation period, which we have, to some degree, already done. Since the observation period in this case includes the periods before, during, and after the recession, the hyperparameters describing dynamic heterogeneity can be used to shed light on how preferences changed, potentially as a result of the economic

[11] For the full set of plots, see the appendix.

[12] There are several caveats to this population-level analysis, which may limit its interpretability or generalizability, and which also limits its comparability to previous studies, e.g. Gordon et al. (2013). Importantly, in this work, we only modeled choice conditional on the purchase decision, and do not capture effects like stockpiling. We also use a relatively lenient rule for retaining consumers in the panel, such that consumers that purchased at least five times were included. This means our estimates of average price elasticities may be subject to panelist attrition.

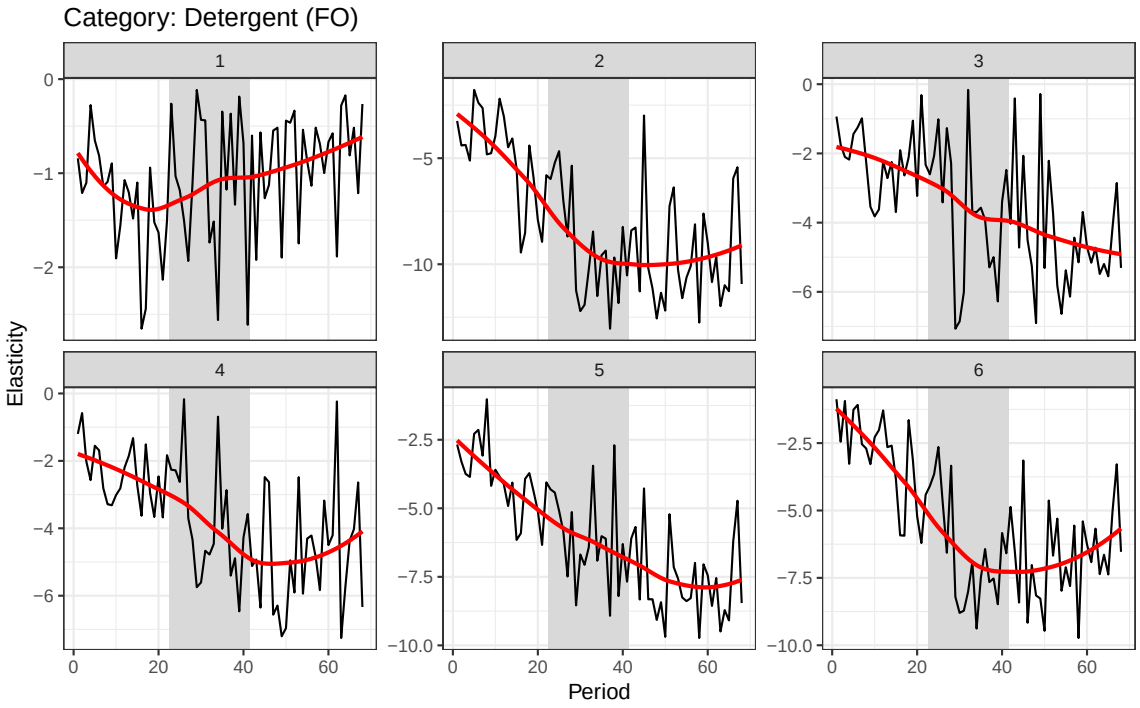

Figure 2.18: The average price elasticity of demand across detergent brands over time, as estimated by the DHGP logit model. The recession era, as defined by NBER, is marked by the grey rectangle. Overlaid on the estimated average price elasticities is a local linear smoothing (LOESS).

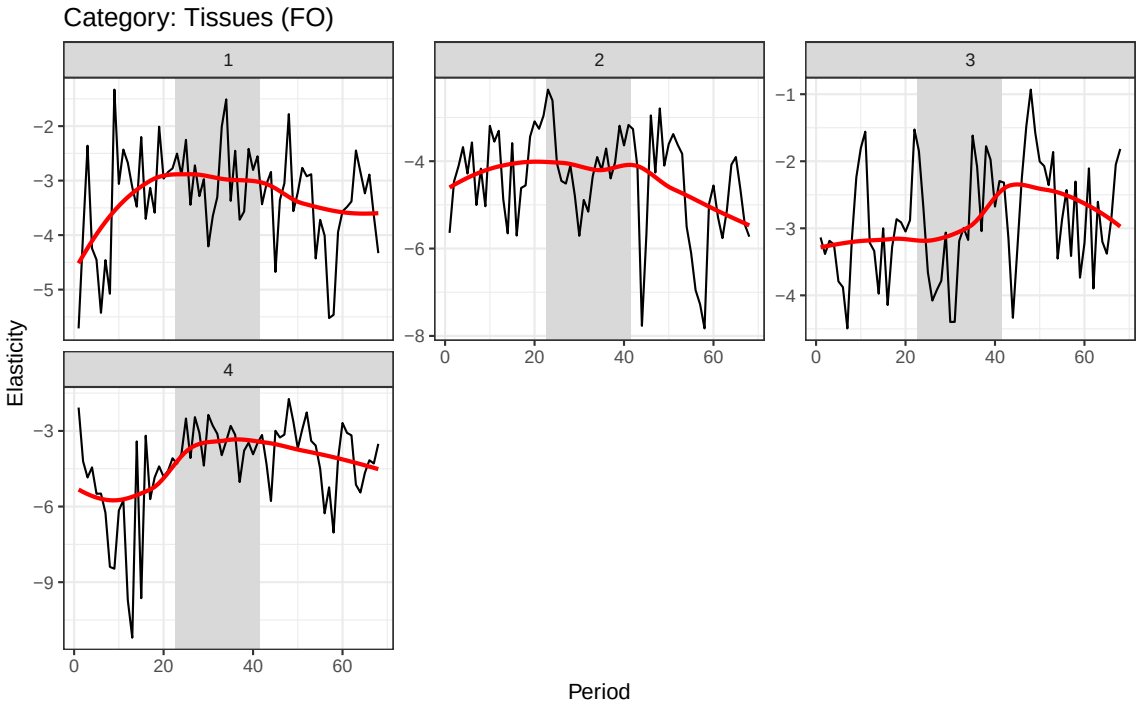

Figure 2.19: The average price elasticity of demand across tissue brands over time, as estimated by the DHGP logit model. The recession era, as defined by NBER, is marked by the grey rectangle. Overlaid on the estimated average price elasticities is a local linear smoothing (LOESS).

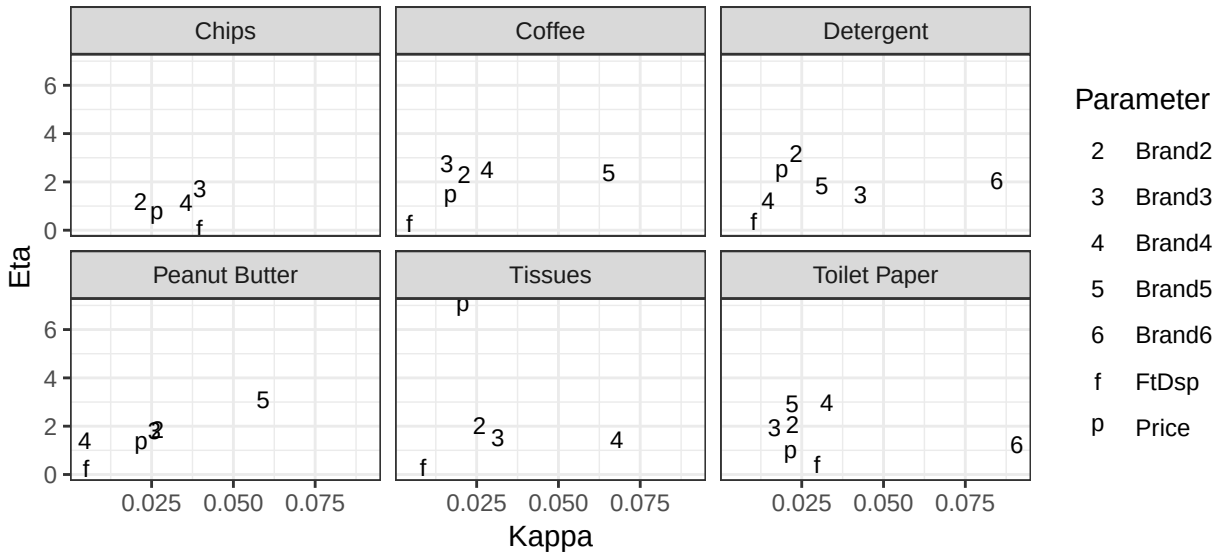

Figure 2.20: Visualization of the mean posterior hyperparameter values across categories.

downturn.

In Figure 2.20, we break down the posterior mean estimates of the hyperparameters by category and coefficient. As described previously, we find that, across categories, it is brand intercepts that typically have the largest values of κ and η. One caveat to this is that the scale of the corresponding variables is different across intercepts and price/feature: intercepts are binary indicator variables, whereas price and feature/display are standardized continuous variables. Hence, interpreting the relative values of η is not straightforward. However, the values of κ are directly comparable, as they relate to the time scale, not the predictor scale. We find that the individual-level intercept parameters exhibited more variation over time, relative to the population curve, than the other coefficients. This is interesting, insofar as it suggests that much of the individual-level dynamics during the recession were driven by shifts among brand preferences, rather than shifts in price or promotion sensitivity directly. This appears to be especially true for lower market share brands, which both tended to have the highest values of κ and η across all brand intercepts.

Individual-level Recession Dynamics

Finally, we can use our individual-level estimates to identify interesting individual-level dynamics. In particular, we can ask, to what degree did shifts in individual-level preferences, relative to the population, align with the recession? That is, to what degree did changes in the distribution of individual-level effects around the mean appear to correspond with the recession? To answer those questions, we conduct two analyses. In the first, we ask the related question: in which period did the curves of individual consumers appear to change the most, relative to the population? In the second, we ask: for individuals who went from one extreme of the distribution to the other (i.e. for crossover cases), at which point did their curves cross over the mean curve?

Individual-level Maximal Rates of Change The goal of this analysis is to isolate periods in which the distribution of heterogeneity around mean appeared to change most dramatically. To measure this, we consider the differenced individual-level estimates, as given by Equation 2.21 and displayed for the tissues category in the bottom of Figure 2.10. The estimates of Diff_{ip} give a notion of how each individual changed relative to the population over time. To isolate periods in which individuals changed most dramatically relative to the population, we then consider the derivative of Diff_{ip}, which we approximate through the slope of locally linear regressions. Finally, for each consumer, in each category, we select the period in which the absolute value of this numeric derivative is highest. Mathematically, this procedure approximates finding:

$$t^*_{ip} = \arg\max_t \frac{d}{dt}\text{Diff}_{ip}(t) = \arg\max_t \frac{d}{dt}\left[\hat{\beta}_{ip}(t) - \mu_p(t)\right]. \tag{2.24}$$

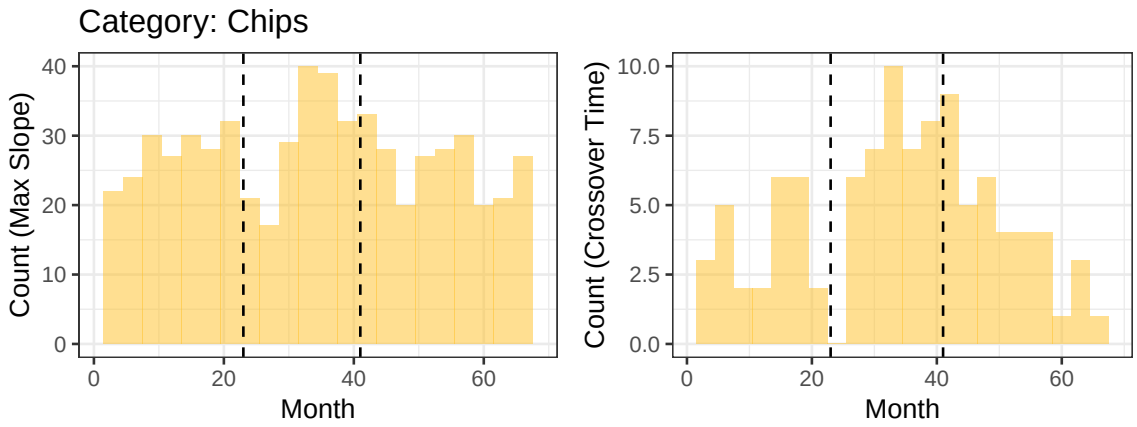

Figure 2.21: At left, the distribution of the timings of maximal slopes for individual-level curves in the chips category, with the recession bounded by the dashed lines. At right, the distribution of the timings of crossovers in the chips category, again with the recession bounded by dashed lines.

From this numeric procedure, we retain only those cases which exhibited significant variation (estimated slope > 0.05), effectively isolating the periods of maximal variation, for cases where there was significant variation. The distribution of the timing of these maximal rates of change then serves as a metric by which we can assess the timing of distributional shifts in preferences.

Timing of Crossovers The second metric we use to assess shifts in the distribution of preferences is by isolating the timing of crossovers: that is, the periods in which individual-level curves crossed over the mean curve. This point is significant insofar as it signifies the period at which a given individual went from the bottom part (half) of the distribution to the top part (half), or vice versa.[13] The distribution of the timings of crossovers serves as another metric by which we can assess in which periods preferences appear to have been changing in interesting ways.

[13]In the case of symmetric marginal distributions, which we often find, "bottom part" is equivalent to "bottom half."

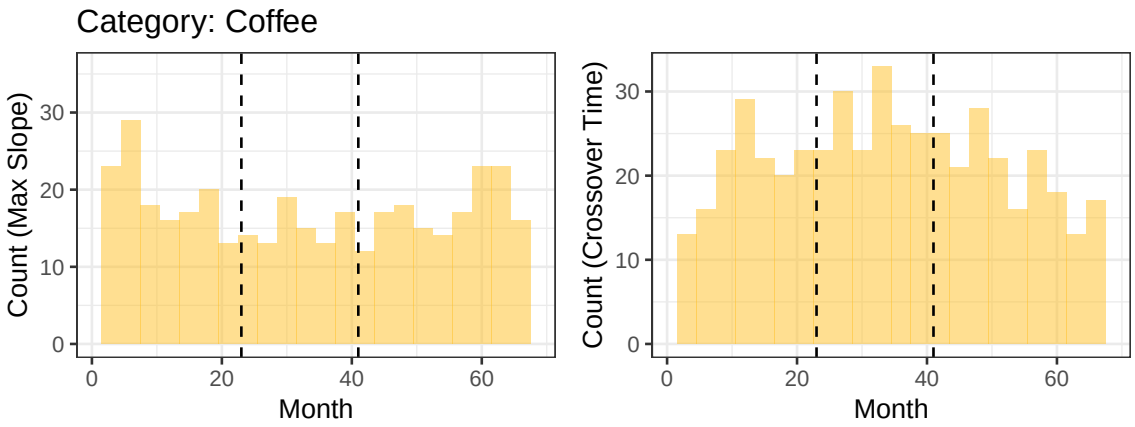

Figure 2.22: At left, the distribution of the timings of maximal slopes for individual-level curves in the coffee category, with the recession bounded by the dashed lines. At right, the distribution of the timings of crossovers in the coffee category, again with the recession bounded by dashed lines.

Results Again, we find an apparent impact of the recession on individual-level dynamics and shifts in the distribution of heterogeneity. Moreover, this impact again seems to be moderated by category. Interestingly, the categories that appear to have the most individual-level shifts associated with the Great Recession are not the same as those that experienced the biggest decrease in price elasticity. In Figure 2.21, for instance, we plot the result for the chips category, where there are striking peaks in both metrics associated with the beginning and the end of the recession. Similarly, we find evidence of such peaks in tissues, which in the elasticity-based analysis, appeared to be countercyclical. In other categories, most notably coffee, we find no evidence of a recession era effect, as shown in Figure 2.22. In fact, in coffee, as well as in detergent (which was found in the elasticity-based analysis to be the most procyclical), the most rapid changes in the distribution of parameters appears to be concentrated toward the ends of the observation window.

2.5 Conclusion

In this paper, we have developed a novel framework for modeling dynamic heterogeneity using Bayesian nonparametric Gaussian process priors and illustrated it in the context of discrete choice models. Our doubly hierarchical Gaussian Process specification fills a void in existing methodology by flexibly allowing for the evolution of parameters at the individual-level, with sharing of statistical information across both individuals and time periods. We have used simulations to show how employing static heterogeneity around dynamic models, although common in the marketing literature, can result in either biased estimates of preference dynamics. In an application to CPG data, we have shown both the prevalence and relevance of dynamic heterogeneity, and the superiority of our framework relative to existing benchmarks. Our application also unearths interesting differences in individual-level preference trajectories during the Great Recession. Finally, we have illustrated the clear managerial implications of considering DHGP-based dynamic heterogeneity, both for understanding preference dynamics, and for developing targeted marketing strategies and setting optimal targeted prices.

While our work clearly highlights the benefits of dynamic heterogeneity, it leaves open several avenues for future research. Foremost, we have illustrated the benefits of dynamic heterogeneity in a choice modeling context. However, the potential applicability of DHGP extends to other modeling contexts involving panel data in which individual-level parameter dynamics are relevant. Moreover, we also limited our exploration of the class of DHGP models to the Matérn kernel at the individual-level, and a small set of mean models at the population-level. There are numerous other possibilities that could be pursued, including richer state-space models for the population dynamics, and more expressive kernels at the

individual-level (e.g. Wilson, 2014). We look forward to these and other extensions, as researchers build on our work, and incorporate dynamic heterogeneity in different application areas.

2.6 Appendix: Extended Fit Statistics

In this section, we present more fit statistics. As a whole, all fit statistics imply that GPDH heterogeneity significantly outperforms statistic heterogeneity, given the same mean model. In the main body of the paper, we presented hit rates in Figure 2.13. In this appendix, we also plot in Figure 2.23 the Watanabe-Akaike Information Criterion (WAIC), which is a Bayesian measure that measures model fit, penalizing for model complexity. We see that this measure again supports the idea that dynamic heterogeneity, as captured through GPDH, better describes the data, even taking into account the added complexity of the model. Interestingly, we find little difference in fit across mean models, except for a noted decrease in fit for the restrictive parametric model.

We also include here the full set of fit statistics, averaged across mean models, for all categories and heterogeneity specifications, in Table 2.5. Those statistics are based on the following counts, for a given brand b:

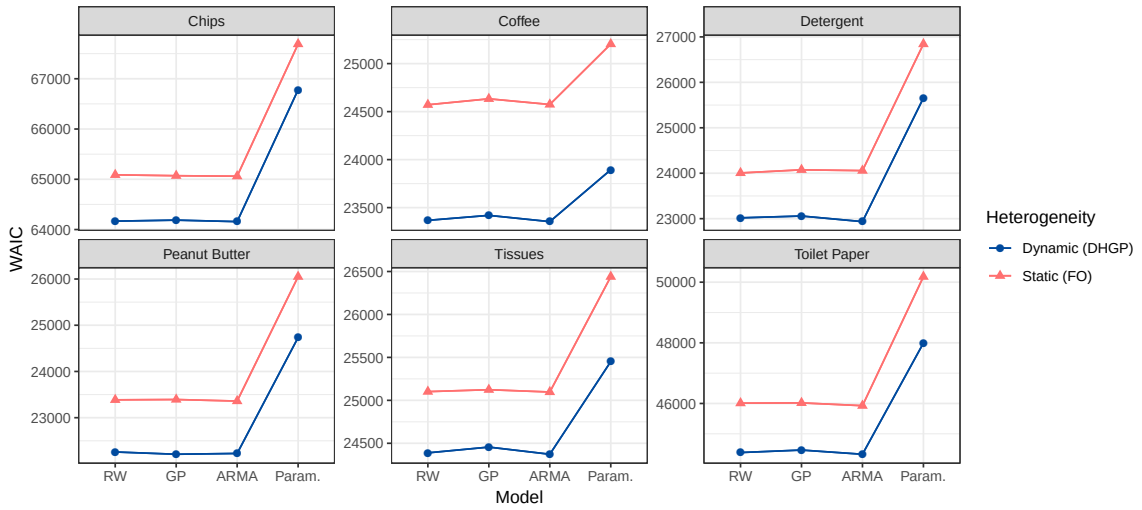

Figure 2.23: WAIC across model specifications. Lower indicates better fit, taking into account model complexity.

- True positives (TP_b) = the number of observations where the model predicted the consumer would choose brand b, and the consumer chose brand b
- False positives (FP_b) = the number of observations where the model predicted the consumer would choose brand b, but the consumer did not choose brand b
- True negatives (TN_b) = the number of observations where the model did not predict the consumer would choose brand b, and the consumer did not choose brand b
- False negatives (FN_b) = the number of observations where the model did not predict the consumer would choose brand b, but the consumer chose brand b.

From these, we compute the following statistics:

- Precision (Prec) - also called the hit rate, equal to $TP_b/(TP_b + FP_b)$
- Sensitivity (Sens) - also called recall or the true positive rate, equal to $TP_b/(TP_b + FN_b)$
- Specificity (Spec) - also called selectivity or the true negative rate, equal to $TN_b/(TN_b + FN_b)$

Finally, we average these across brands in the following ways:

- Macro average: the average of each of the above rates. Intuitively, this aggregation treats all classes equally, ignoring potential class imbalance.
- Micro average: this aggregation computes the above statistics by summing over b at each step. Intuitively, this takes into account class imbalance, at the risk of showing good performance when one class dominates.

In-sample

		Macro			Micro			Max			Min		
Category	Heterogeneity	Prec	Sens	Spec	Prec	Sens	Spec	Prec	Sens	Spec	Prec	Sens	Spec
Chips	GPDH	0.695	0.629	0.882	0.695	0.695	0.898	0.709	0.832	0.978	0.665	0.484	0.724
Chips	FO	0.665	0.599	0.873	0.671	0.671	0.890	0.689	0.817	0.975	0.631	0.434	0.706
Coffee	GPDH	0.782	0.754	0.940	0.780	0.780	0.945	0.804	0.844	0.985	0.756	0.675	0.879
Coffee	FO	0.729	0.696	0.926	0.726	0.726	0.931	0.763	0.805	0.983	0.673	0.608	0.854
Detergent	GPDH	0.836	0.813	0.967	0.843	0.843	0.969	0.868	0.918	0.992	0.796	0.732	0.927
Detergent	FO	0.801	0.774	0.960	0.811	0.811	0.962	0.844	0.905	0.991	0.717	0.665	0.913
Peanut Butter	GPDH	0.832	0.819	0.956	0.830	0.830	0.958	0.874	0.872	0.984	0.810	0.749	0.929
Peanut Butter	FO	0.789	0.776	0.946	0.792	0.792	0.948	0.858	0.843	0.982	0.717	0.632	0.911
Tissues	GPDH	0.762	0.751	0.916	0.761	0.761	0.920	0.776	0.788	0.970	0.741	0.703	0.866
Tissues	FO	0.717	0.704	0.901	0.718	0.718	0.906	0.735	0.752	0.963	0.701	0.630	0.840
Toilet Paper	GPDH	0.791	0.781	0.957	0.792	0.792	0.958	0.818	0.846	0.984	0.742	0.710	0.924
Toilet Paper	FO	0.746	0.736	0.948	0.750	0.750	0.950	0.789	0.818	0.981	0.709	0.665	0.911

Forecast

		Macro			Micro			Max			Min		
Category	Heterogeneity	Prec	Sens	Spec	Prec	Sens	Spec	Prec	Sens	Spec	Prec	Sens	Spec
Chips	GPDH	0.646	0.528	0.865	0.647	0.647	0.882	0.777	0.769	0.991	0.524	0.182	0.704
Chips	FO	0.618	0.527	0.863	0.643	0.643	0.881	0.778	0.758	0.987	0.434	0.200	0.708
Coffee	GPDH	0.631	0.629	0.904	0.652	0.652	0.913	0.756	0.718	0.967	0.522	0.521	0.812
Coffee	FO	0.615	0.607	0.900	0.634	0.634	0.909	0.753	0.699	0.969	0.492	0.521	0.810
Detergent	GPDH	0.717	0.618	0.947	0.764	0.764	0.953	0.878	0.931	0.996	0.453	0.177	0.858
Detergent	FO	0.720	0.611	0.943	0.751	0.751	0.950	0.877	0.910	0.996	0.502	0.218	0.833
Peanut Butter	GPDH	0.519	0.545	0.886	0.547	0.547	0.887	0.752	0.650	0.950	0.331	0.332	0.822
Peanut Butter	FO	0.492	0.520	0.879	0.526	0.526	0.882	0.673	0.643	0.931	0.312	0.279	0.820
Tissues	GPDH	0.558	0.559	0.846	0.545	0.545	0.848	0.712	0.735	0.924	0.459	0.470	0.768
Tissues	FO	0.541	0.539	0.840	0.529	0.529	0.843	0.699	0.706	0.923	0.445	0.458	0.759
Toilet Paper	GPDH	0.588	0.574	0.920	0.613	0.613	0.923	0.747	0.782	0.978	0.369	0.239	0.859
Toilet Paper	FO	0.562	0.550	0.917	0.598	0.598	0.920	0.722	0.813	0.980	0.304	0.169	0.852

Table 2.5: Fit statistics average across mean model. The statistics are described above.

- Max: the max over b. Intuitively, this is the statistic for the class that was easiest to predict.

- Min: the min over b. Intuitively, this is the statistic for the class that was most difficult to predict.

2.7 Appendix: Average Elasticity Plots

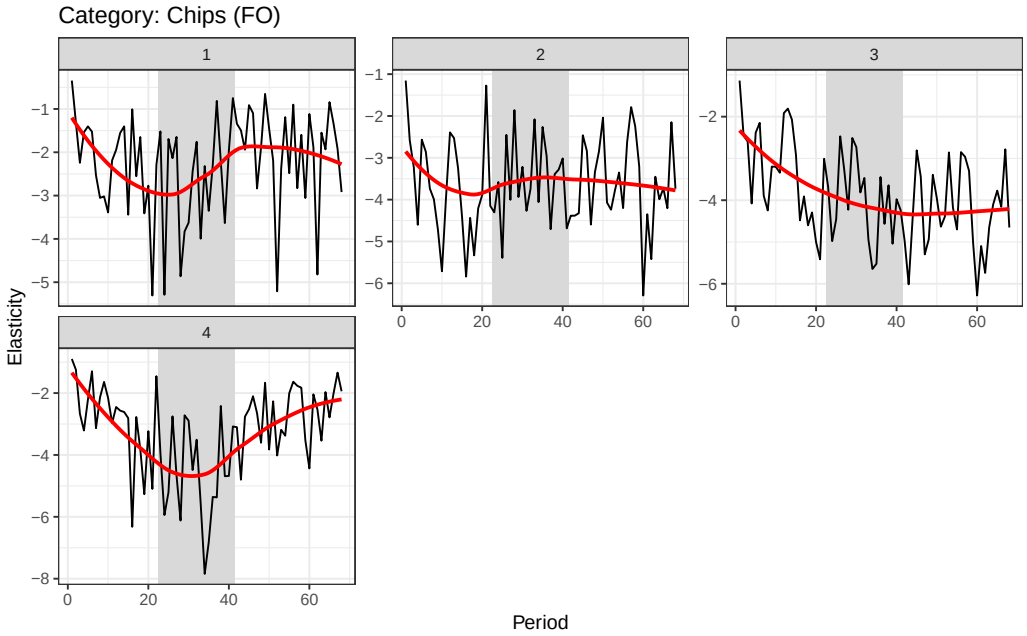

Figure 2.24: The average price elasticity of demand across chips brands over time, as estimated by the DHGP logit model. The recession era, as defined by NBER, is marked by the grey rectangle. Overlaid on the estimated average price elasticities is a local linear smoothing (LOESS).

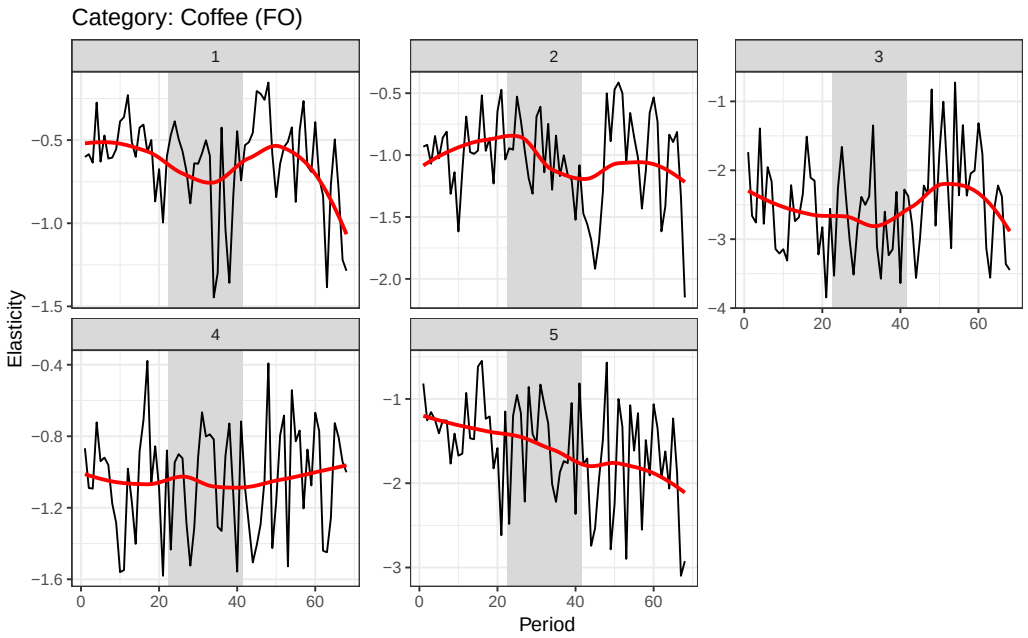

Figure 2.25: The average price elasticity of demand across coffee brands over time, as estimated by the DHGP logit model. The recession era, as defined by NBER, is marked by the grey rectangle. Overlaid on the estimated average price elasticities is a local linear smoothing (LOESS).

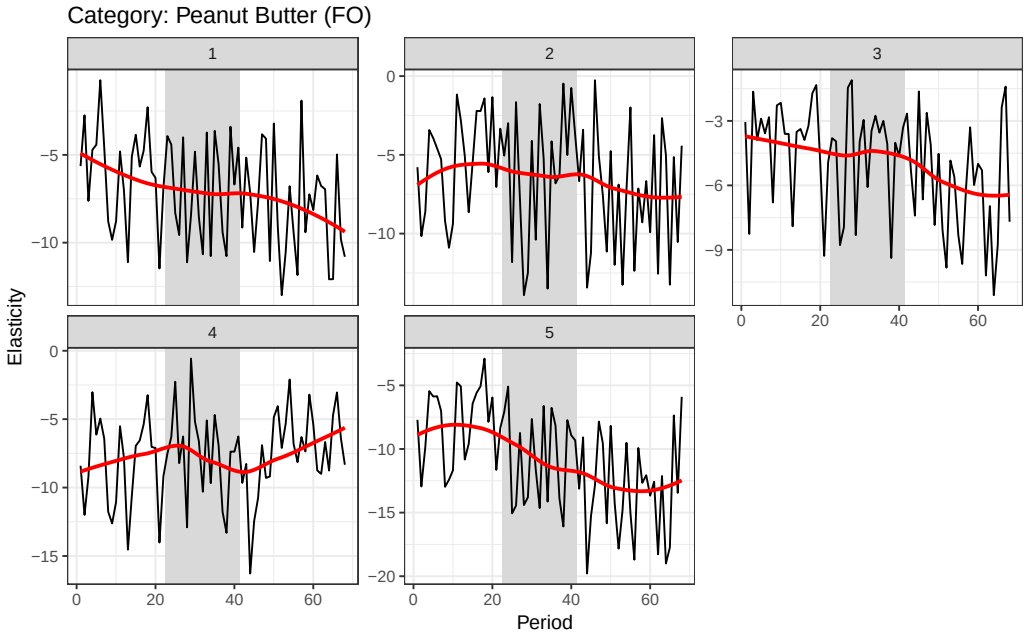

Figure 2.26: The average price elasticity of demand across peanut butter brands over time, as estimated by the DHGP logit model. The recession era, as defined by NBER, is marked by the grey rectangle. Overlaid on the estimated average price elasticities is a local linear smoothing (LOESS).

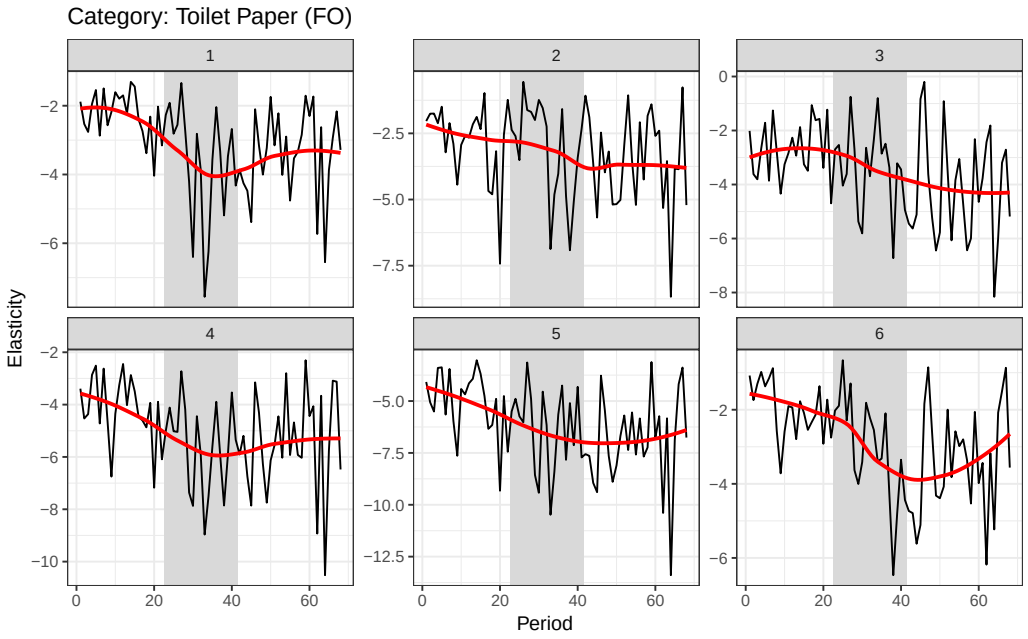

Figure 2.27: The average price elasticity of demand across toilet paper brands over time, as estimated by the DHGP logit model. The recession era, as defined by NBER, is marked by the grey rectangle. Overlaid on the estimated average price elasticities is a local linear smoothing (LOESS).

2.8 Appendix: Curve Timing Plots

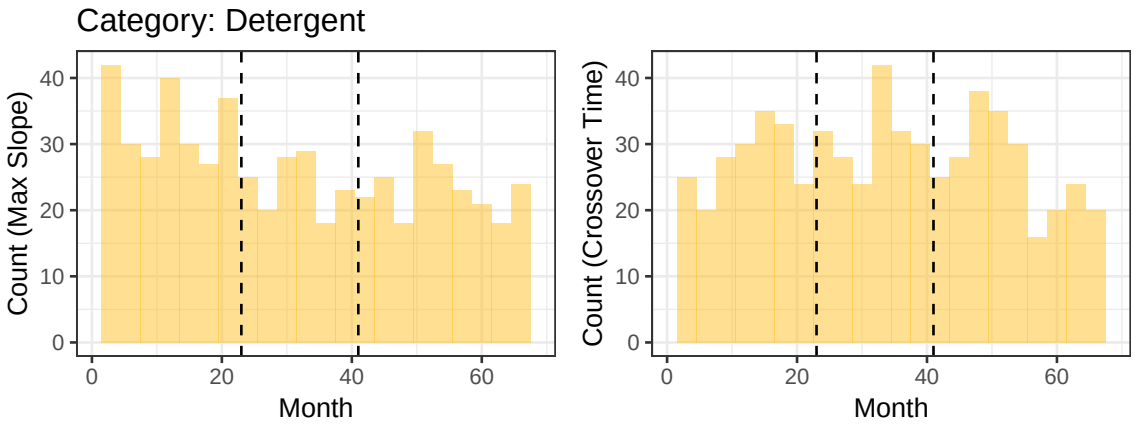

Figure 2.28: At left, the distribution of the timings of maximal slopes for individual-level curves in the detergent category, with the recession bounded by the dashed lines. At right, the distribution of the timings of crossovers in the chips category, again with the recession bounded by dashed lines.

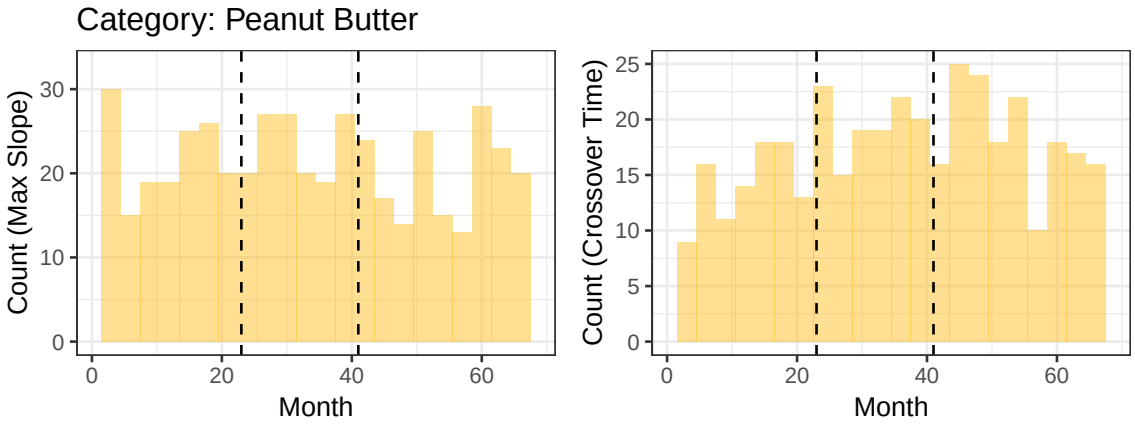

Figure 2.29: At left, the distribution of the timings of maximal slopes for individual-level curves in the peanut butter category, with the recession bounded by the dashed lines. At right, the distribution of the timings of crossovers in the coffee category, again with the recession bounded by dashed lines.

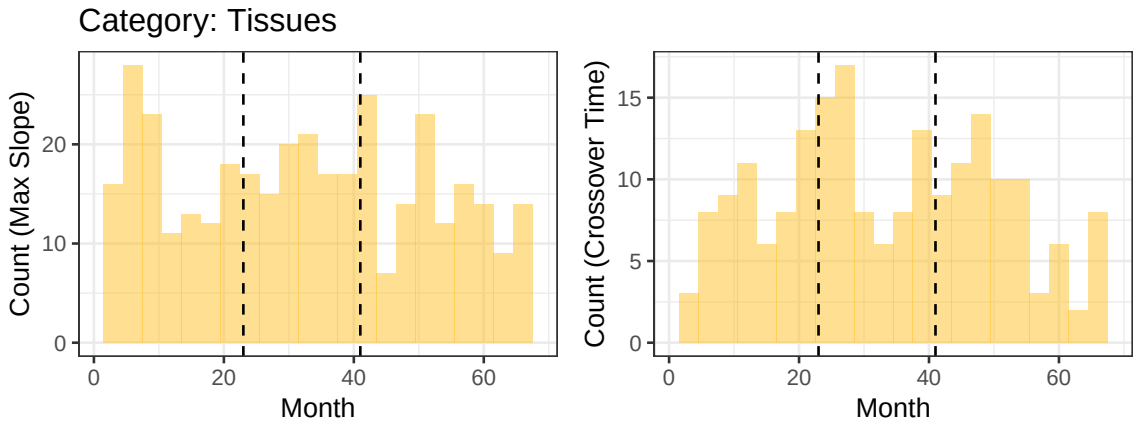

Figure 2.30: At left, the distribution of the timings of maximal slopes for individual-level curves in the tissues category, with the recession bounded by the dashed lines. At right, the distribution of the timings of crossovers in the chips category, again with the recession bounded by dashed lines.

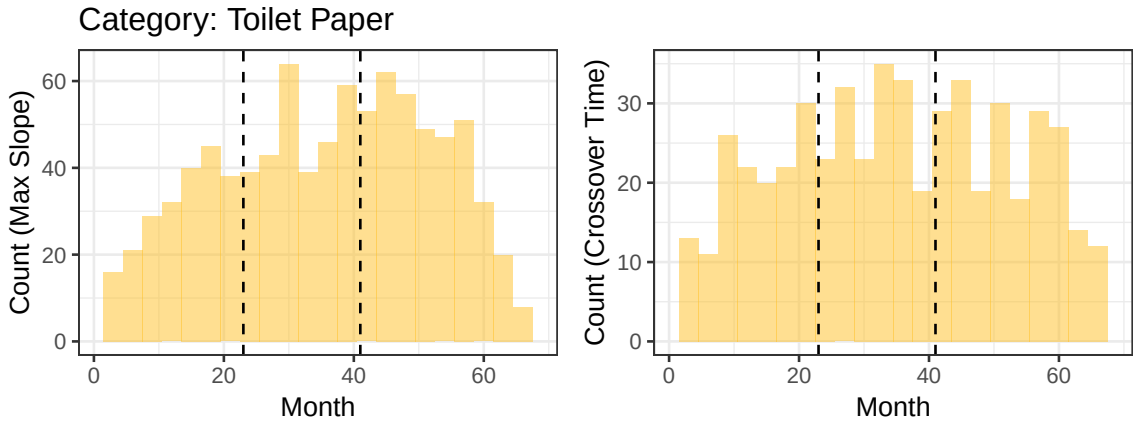

Figure 2.31: At left, the distribution of the timings of maximal slopes for individual-level curves in the toilet paper category, with the recession bounded by the dashed lines. At right, the distribution of the timings of crossovers in the coffee category, again with the recession bounded by dashed lines.

3

Letting Logos Speak: A Machine Learning Approach to Data-Driven Logo Design

This essay forms the basis of a working paper of the same name, which is in preparation for submission. That paper is jointly authored with Asim Ansari and Olivier Toubia.

Abstract

Logos serve a fundamental role in branding as the visual figurehead of the brand. Yet, due to the difficulty of using unstructured image data, prior research on logo design has been largely limited to non-quantitative studies. In this work, we explore logo design from a data-driven perspective. In particular, we aim to answer several key questions: first, to what degree can logos represent a brand's personality? Second, what are the key visual elements in logos that elicit brand and firm relevant associations, such as brand personality traits? Finally, given text describing a firm's brand or function, can we suggest features of a logo that elicit the firm's desired image? To answer these questions, we develop a novel logo feature extraction algorithm, that uses modern image processing tools to decompose unstructured pixel-level image data into meaningful visual features. We then analyze the links between firm identity and the features of its logo, through both predictive modelling and a deep, multiview generative model, which links visual features of logos with textual descriptions of firms and consumer ratings of brand personality by learning representations of brand identity. We apply our modeling framework on a dataset of hundreds of logos, textual descriptions from firms' websites, third party descriptions of firms, and consumer evaluations of brand personality to explore these questions.

3.1 Introduction

Logos are the most distinct marks of brands, adorning everything from packaging to advertising. Designers create logos to represent the essence of brands, and firm's motivate brand and logo redesigns with an intention to convey a new idea. Yet, despite the overwhelming significance of logos, and the substantial costs of logo redesigns, marketing scholars have paid relatively little attention to the logo design process.

In this work, we show that there is a science to the logo design process that can be captured by models, and that such models can serve as a basis for understanding the meaning conveyed by logos, as well as aid brands in the design of logos consistent with their brand identities. In particular, we synthesize novel image processing techniques with machine learning models for prediction and multi-view learning to capture the links between a brand's function and personality, and its logo features.

Our work makes several contributions. Foremost, it is the first paper to study logos from a holistic and quantitative perspective. This is important, first, because it adds a level of objectivity to the design process: while our model cannot replace the creative touch of designers, it can offer both designers and firms guidance in crafting their brand identities, in an objective fashion. When weighing competing designs and opinions, an objective prediction of the reactions of consumers to a logo design can allow managers to make a data-driven decision, in what has historically been viewed as a creative domain. A model-based approach lets us simulatenously assess the many facets of logo design, and explicitly measure the impact of the overall design of the logo. Finally, because a model-based approach allows us to make design recommendations, it can be used even by budget

strapped firms to thoughtfully design their logos in a data-driven fashion, which, as we will show, can be a determinant of branding success.

From a methodological perspective, ours is among the first papers in marketing to directly use image data, without relying on human coders. Specifically, our work presents a novel approach to working with unstructured, visual data, through a theory-driven image processing approach. Specifically, our feature extraction algorithm decomposes logos into meaningful features, many of which are driven by prior theory about how logos convey meaning. The set of these features forms a "visual dictionary" which we can use to describe logos in a way that is meaningful to designers, and that is also amenable to probabilistic modeling. Working directly with image data is important for wide and general applicability of our framework, as well as for scalability: for brand managers or designers to use our model in practice, it cannot be based on the inputs of human coders.

Our work is also among the first in marketing to synthesize both unstructured text and image data. The model we develop for that purpose is called a multiview variational autoencoder. Variational autoencoders (VAE) are popular machine learning tools for learning representations of complex data. In this work, we develop a multimodal variational autoencoder, which learns representations of brands across all of the ways in which brand is manifest: text, logo, and brand personality. The task of learning unified representations across domains is an instance of multiview or transfer learning. As we operationalize transfer learning via learned representations that are shared across domains, it is also an instance of representation learning (Li et al., 2016). While we largely draw on standard representation learning methodologies (i.e. VAE), our inference algorithm features a novel multimodal encoder structure, which is tailored to the task of transfer learning across complementary domains. The result is a set of functions that can

predict across domains: for example, given a textual description of the brand, what features do we expect to find in that brand's logo, and how do we expect consumers to perceive the brand's personality?

The rest of the paper is organized as follows: in Section 3.2, we review the existing literature on logo design and aesthetics in marketing. In Section 3.3, we describe the unique dataset we have compiled to calibrate our model. In Section 3.4, we briefly describe how images are stored at the data-level, then describe our logo feature extraction algorithm. In Section 3.5, we present descriptive and "model-free" predictive evidence of the links between design, brand personality, and firm function. In Section 3.6 , we develop a multi-view learning model of brands and their logos, and in Section 3.7, we show the results of applying that model to our data, including examples of the learned representations, logo recommendations, and links to brand personality. Finally, we conclude with a summary of on-going research and directions for further study.

3.2 Literature

There is a sizable literature, especially in consumer behavior, on how consumers react to aesthetics, both in logos and in other aspects of marketing. Some of this literature describes important dimensions for logos, and studies how they correlate with or predict consumer-level outcomes. Other papers discuss how these reactions vary cross-culturally, or the mechanisms governing consumers' reactions to various visual stimuli. In this section, we review those findings, with an eye toward informing our own model of logo design, with respect to important dimensions of logo design, and what features have been shown to have a measurable impact on consumers.

3.2.1 Logos

A limited amount of research in marketing has been done specifically on firm logos, starting with Henderson and Cote (1998), where they use factor analysis on a set of logo traits, coded by experts, to come up with a set of constructs that describe logos generally: natural, harmonious, elaborate, parallel, repetition, proportion, and roundness. Of their factors, only natural, harmonious, and elaborate (from now denoted NHE) seem predictive of outcome measures generally. In Henderson et al. (2003), they test whether these constructs hold cross-culturally, finding little difference of the predictive power of NHE in Asia versus the United States. This cross-cultural work is then expanded by van der Lans et al. (2009), again using NHE, together with three "objective design measures"—repetition, proportion, and parallelism, all determined by expert coders from disparate geographies. They find the NHE dimensions are universally good descriptors of design, even cross culturally. Together, these studies support the idea that NHE provide a good proxy for design elements of logos.

Other work has looked at specific aspects of logos. Klink (2003), for example, studies the link between the brand name and the traits of the logo, finding ties between the phonetic structure of the name and the traits used in the logo, such as color and angularity. Walsh et al. (2010) find that moving from an angular logo to a round logo produces generally mixed responses in consumers, dependent on their level of commitment to the brand. The idea of circular versus angular logos is also explored in Jiang et al. (2015), where they find that the mere circularity or angularity of the logo affects perceptions of the product and the company, through perceived hardness or softness, which in turn influences attribute judgments. Other studies look at the orientation of the logo, including Cian et al. (2014), where they

find that different logos can evoke the idea of movement, often through the positioning of the logo elements or the horizontal orientation of the logo, which in turn affects consumers' engagement and attitudes. Even more recently, Schlosser et al. (2016) find that upward diagonals convey greater activity than downward diagonals, leading to more favorable product evaluations, greater efficacy beliefs, and greater post-consumption satisfaction. Together, these studies imply that among the objective design measures employed in a design model should be traits like color, angularity, and orientation.

Finally, there has been a significant amount of work done on typeface and font. Doyle and Bottomley (2006) provide an excellent overview and study of fonts in logos, describing both the background of typeface research, and studying specifically the appropriateness of a given typeface for describing a particular product or brand. They define appropriateness in terms of abstract connotations,[1] where abstract connotation is captured by Osgood's evaluation, potency, and activation dimensions (EPA), a set of factors that has been shown across contexts (including typeface) to capture abstract connotations. They find that congruence in EPA between the font and the product leads to more frequent choice of the product. In another study, Hagtvedt (2011) shows that incomplete typeface can lead to both perceptions of untrustworthiness and increased innotvativeness. Hence, an understanding of the role of font also seems important.

3.2.2 Aesthetics

While academic work specifically on logos has been relatively limited, there is a large body of work on aesthetics and perception, some of it in marketing, especially

[1]Abstract connotations differ from, for example, direct connotations, like, for example, a font with "snowcaps" being associated with something cold

in the domain of consumer response to advertising.

Color In marketing, Deng et al. (2010) study consumers' preferences for color combinations in product design. They have three main findings. First, of the three common dimensions of color—hue, saturation, and lightness—they find people tend to de-emphasize lightness, relative to the other two. Second, in product design, people prefer generally similar colors, but with a single contrast color, where the contrasting color is often used to highlight a single distinctive element. Finally, they find that people generally prefer a small number of colors. Kareklas et al. (2014) also explore color in marketing. They find that people exhibit an automatic preference for white over black in product choice and advertising, similar to the implicit bias observed in other studies in psychology. Relatedly, Semin and Palma (2014) find that white is perceived as more feminine, whereas black is perceived as more masculine. In psychology, more work has been done on color. For example, Valdez and Mehrabian (1994) study the effect of color on emotions, finding that of the three key color dimensions, saturation and lightness drive emotional responses along the pleasure, arousal, and dominance dimensions. They also find shades of blue, green, and purple to be the most pleasant, and shades of yellow to be the least pleasant.

Font Besides logos, font and typeface have also been explored both in the domain of advertising, and in impression management generally. Childers and Jass (2002) explores the influence of typeface on perceptions, finding that the semantic connotations of typeface can influence consumers' ratings of products. Henderson et al. (2004) take a different approach and analyze many extant fonts in an effort to summarize their impressions and design features. They come up with a set of four factors—pleasing, engaging, reassuring, and prominent—that describe typeface

impressions, and six factors—elaborate, harmony, natural, flourish, weight, and compressed—that describe typeface design, based on the typology literature and ratings of experts, and conclude that there may be universal design elements that can help managers in impression management.

Orientation In an early study on advertising, Meyers-Levy and Peracchio (1992) show that the camera angle of an ad showing a product can influence consumers' judgments of the product, moderated by processing motivation. Specifically, they find that when processing motivation is low, looking up at the product yields more favorable judgments; alternatively, when processing motivation is moderate, looking at an eye-level product is best. More recently, Chae and Hoegg (2013) find that in cultures where reading is done from left to right, products are viewed more favorably when positioned congruently with this spatial orientation (and vice versa). Deng and Kahn (2016) find that the location of the product image on its packaging (top/left or bottom/right) influences the item's perceived weight (lighter or heavier respectively).

Other A host of other papers discuss other aspects of aesthetics that might be relevant for logo design. For example, Navon (1977) finds that global features are processed more readily and fully than local ones, a trait we might expect to operate also in logos. More recently, Pieters et al. (2010) use eye-tracking to study the visual complexity of advertisements. They come up with two distinct aspects of visual complexity: feature complexity and design complexity. Feature complexity simply refers to variation in basic features like color and edges, and is measured by variance at the pixel level, while design complexity refers to variation in the elaborateness of the design, and is measured by six general principles: quantity of objects, irregularity of objects (shape), dissimilarity of objects, detail of objects,

asymmetry of object arrangement, and irregularity of object arrangement.

Relevant to relating brand constructs to visual elements, Orth and Malkewitz (2008) decompose package design into five distinct "types"—massive, contrasting, natural, delicate, and nondescript—and relate those types prescriptively to brand personalities. In an excellent review article, Spence (2012) discusses fascinating cross-modal effects, including things like the visual perceptions associated with tastes and textures (e.g. the angularity of carbonation or bitterness), which could be relevant determinants of logo design. Spence argues that firms can use these principles to set up an appropriate cross-modal expectation for a consumption experience, thereby enhancing it. This, in turn, is based off earlier work that discusses consumers preferences for congruity in the consumption experience (e.g. a fancy logo matching a fancy experience; see Patrick and Hagtvedt (2011) for an example of this kind of effect).

3.3 Data

To understand the links between logos and brand identity, we have compiled a rich dataset of brands and their logos. Our goal is to understand both what brand-relevant concepts a given logo conveys, and how a firm can design a logo consistent with those concepts. To that end, our dataset consists of four components: logos, textual descriptions of firms from the firms' websites, textual and other descriptors from a third party source, and brand personality ratings from consumers reacting to both the logo and description.

Our insights derive from linking the logos and descriptions of existing brands; hence, we must ensure that the firms we use are high quality, having given

some thought to the design of their logos. As a proxy of this, we chose firms that were either rated as having a strong brand identity by brand specialists, or were highly profitable and recognizable, with the rationale that these firms have likely invested in their brand identity as part of their success. Specifically, we looked at all firms that were either listed in the Interbrand brand consultancy's list of Top 100 Global Brands of 2016, listed as among the top 500 most valuable American brands of 2016 by the brand valuation consultancy Brand Finance, or listed in the Forbes 500 in 2016. There was a large degree of overlap between the lists, leaving us with a final sample of 715 firms.

Logos Firms typically employ a variety of logos for different purposes. Broadly speaking, a logo may be comprised of three key features: marks, logotype, and subtext. Marks are the non-textual parts of the logo (e.g. the Apple apple, or the Nike swoosh); the logotype is the primary textual identifier, usually displaying the brand name; and the subtext is other text, often a brief descriptor of the brand. A logo always has either a mark or a logotype, while some logos have both, and some include a subtext. Some firms employ variants of their logo for different purposes, which may consist of either just the mark, or just the logotype, or the mark and logotype omitting the subtext, or a logo where the colors are inverted (e.g. blue lettering on a white background becomes white lettering on a blue background). Determining which logo to use thus requires some amount of judgment on the part of the researcher. As a rule, we selected logos with white backgrounds, if such a logo is in use. Similarly, we selected the logo with both logotype and mark, if it is in use by the firm. For other aspects of the logo, including subtext and the orientation of the mark relative to the logotype, we used the version that appeared most commonly on the firms online marketing materials.

Text We collected textual descriptions of two sorts: first, we collected *web descriptions*, consisting of both functional and brand-relevant text taken directly from firms' websites. We collected this data in two batches: in one, we asked Amazon mechanical turk users to find text on the firm's website that describes how the firm views its brand, and that does *not* merely describe what the firm does. We guided workers toward the About Us, Mission Statement, Corporate Values, or Investor Relations pages of firms' sites. In a second batch, we asked workers to find text that describes what the firm does, and is not identical to the text already supplied. In both cases, we gave incentives for workers to provide long descriptions. We further ensured that each description was of minimal quality, and at least one paragraph (three sentences) in length.

In order to understand the firm from a third party perspective, we also used textual descriptions of firms and other tags from the database *Crunchbase*. Crunchbase is commonly used by investors to learn about firms. As such, it contains relatively straightforward and consistent descriptions of what firms do.

Brand Personality Finally, we also collected *brand personality ratings* (Aaker, 1997). Specifically, we again used Amazon Mechanical Turk to gather the ratings. To elicit the ratings, we showed U.S.-based participants on Amazon Mechanical Turk both the logo and text describing the firm. Then, we asked participants to rate the extent to which they thought each of a set of traits describes the focal firm, based on the logo and text provided. We used the original set of 42 personality traits from Aaker (1997), as well as three reverse-coded attention check traits[2] We gathered 20 responses per brand.

[2]The reverse-coded traits were honest/dishonest, exciting/boring, and good-looking/ugly. Any participant who answered that both traits are descriptive of the firm was automatically removed.

3.4 Logo Feature Extraction

The primary barrier to using visual data in models is the difficulty of working with unstructured image data. Many methods have been developed for incorporating images in models, with much of the literature coming from the computer vision and machine learning communities. Broadly, there are two approaches to using images in models: the first uses raw pixel-level data as the input to a probability or machine learning model. This is common, for example, in models of image recognition or image captioning, where the model is typically based on a neural network, and the task is a supervised prediction task. A second approach first processes the image, then uses the outputs from this processing as an input to the model. A common approach here is to create an image "dictionary" of representative image features, including the common feature detection algorithms like scale invariant feature transform (SIFT) (Lowe, 1999).

In our work, we follow the second approach: to incorporate the logos into our model of design, we first process the logo image into logo features, through a novel logo feature extraction algorithm based on modern image processing methods. Unlike common algorithms like SIFT, which isolates features of images optimized for use in computer tasks (e.g. scale invariance for image recognition), our algorithm aims to distill a logo into components that are meaningful for consumers and designers, rooted in the literature on logos and aesthetics.

3.4.1 Algorithm Overview

Our algorithm has three general stages: in the first stage, which we term summarization, we compute a variety of features from the logo as a whole, which we

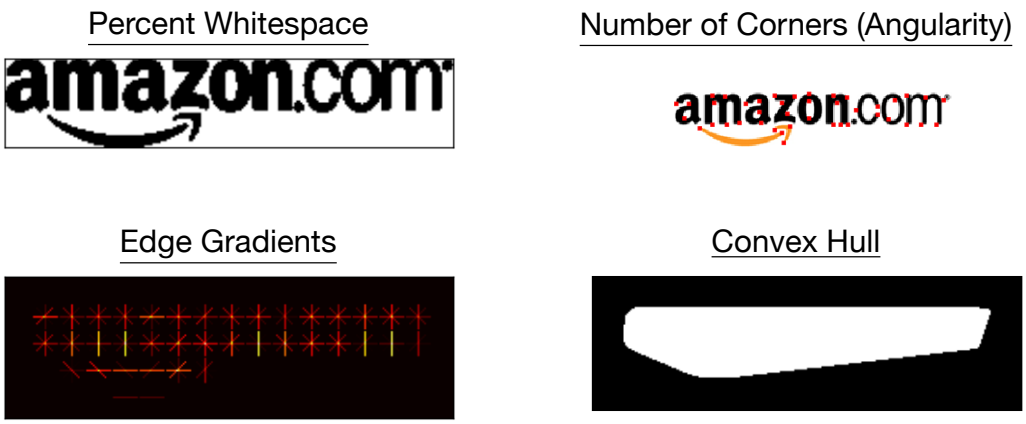

Figure 3.1: Examples of global features, using Amazon's logo as an example. Percent whitespace captures the percentage of pixels that are white (background), within the convex hull of the logo. The number of corners is a measure of angularity computed via the Harris corner detector. Edge gradients capture directionality of edges in the logo, and are computed by computing numerical gradients sliding over a binarized (black and white) version of the logo. The convex hull is the smallest convex polygon containing all of the non-background pixels.

refer to as global summary features. Examples of these features are given in Figure 3.1, using Amazon's logo as an example. One such computation is a density-based color quantization, where we learn how many distinct colors are in each logo. In the second stage of the algorithm, which we term segmentation, we assign each pixel in the logo to one of these colors, then segment the logo into regions that are separated either by color or by background (i.e. the color white). For each of these segments, we separate them into characters and marks. This process is illustrated in Figure 3.2, again using Amazon's logo as an example. In the final stage, which we term tokenization, we cluster several of the features across logos, including the color, hull shape, and mark shape, to form a dictionary describing common classes of features. We describe these features in the remainder of this section, and leave the details of this process to the subsequent section, and to the appendix.

Figure 3.2: Examples of the segmentation process, using Amazon's logo as an example. The original logo is at top. Beneath that is the segmented logo, where black identifies the background, and distinct regions are marked by different color regions. We then apply a template matching and filtering algorithm to identify which of these regions are characters (bottom-right), and assume the remainder are the marks (bottom-left).

3.4.2 Visual Features

A comprehensive listing of all of our visual features can be found in the table in Appendix 3.9, including descriptions on each feature, and links to the literature. In general, the features are structured around feature types, which are themselves drawn from the literatures on logos and aesthetics.

Color The full dictionary of colors is given in Figure 3.3. This is computed by clustering colors across all of the logos in the dataset. Besides for just computing which colors there are, both in the logo as a whole, and in each mark, we also assess which color is the dominant color, which is an accent color, and how much whitespace there is within the convex hull of all logo pixels. We also compute other summary statistics about color in the hue-saturation-value (HSV) color space, including the mean and standard deviation of the saturation and lightness channels.

Name	R	G	B	Color	Name	R	G	B	Color
White	253	253	253		Dark Blue	30	42	124	
Black	20	18	18		Light Gray	165	164	167	
Red	226	33	41		Light Blue	54	153	204	
Blue	25	89	152		Light Green	99	178	67	
Dark Green	34	120	77		Yellow	245	202	36	
Orange	239	131	40		Tan	186	164	103	
Dark Gray	116	111	111		Dark Red	174	39	63	

Figure 3.3: The color dictionary: This table shows the RGB color channel values of the cluster centers for the representative set of colors, along with the actual color encoded by those values. These were obtained by clustering in the LAB color space across logos, which is meant to capture differences in human color perception.

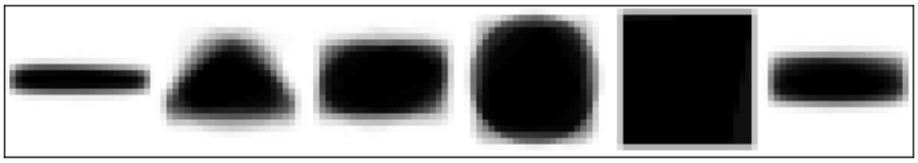

Figure 3.4: The hull classes: This table shows the six typical shapes of logos, as characterized by their convex hulls. Each logo in our dataset is assigned to one of these classes.

Format and Shape These variables include things like whether or not the logo has a mark, what is the size of each mark, how many marks are there, and what is the aspect ratio of the logo (mark). For the logo as a whole, we also compute the convex hull, which is the smallest convex polygon that contains all of the non-background pixels. We then cluster these hulls across logos to form a dictionary of logo shapes, which is shown in Figure 3.4. We also do something similar for the shape of the marks: for each mark, we standardize its shape and convert it to greyscale, then cluster across marks into 14 representative mark types. This maintains more details than the convex hull approach, allowing us to see, for example, the difference between solid and hollow circles, but is also typically more noisy. We give examples of these classes in Figure 3.5.

Font Font is a crucial feature of logos. We therefore have developed an elaborate and extensive procedure to try to identify and describe characters and their fonts.

Cluster	Sample of marks
6	
7	
9	

Figure 3.5: The mark classes: This table shows three examples of our mark classes, with 10 randomly sampled examples of each.

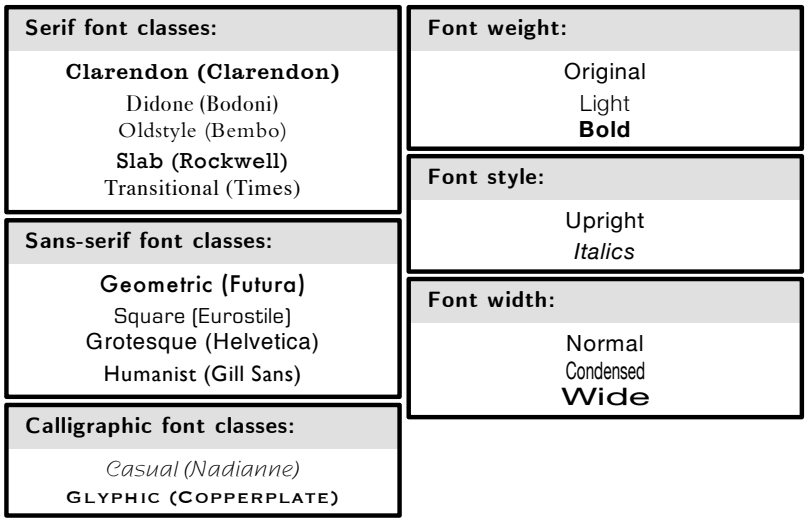

Figure 3.6: Font classification system employed by the algorithm: fonts were matched to a font family, weight, style, and width.

Specifically, for each segment of the logo, we apply a template matching procedure, to try match the segment to an extensive collection of fonts, which we curated to capture the intricacies of font design as exhaustively as possible. This font dictionary captures a range of font families, forms, and stylings, including examples of fonts from all Vox-ATypI font classes, a standard font classification scheme used by font experts.[3] We illustrate our complete font typology in Figure 3.6.

Other There are several other features which have been previously illustrated to be important aspects of design: complexity, symmetry, repetition, and orientation. For each of these, we include direct or indirect measures aimed at capturing that

[3] `https://en.wikipedia.org/wiki/Vox-ATypI_classification`

feature, without the need for a human coder. For complexity, we include a number of measures, including the number of distinct colors, the number of segments, the permetric complexity (the ratio of edge pixels to interior area), and the greyscale entropy (the average variance of pixel intensities across sliding windows). We also include measures of both horizontal and vertical symmetry, computed by looking at the correlation between halves of the image. For repetition, we look at the different subregions of the logo, and compute correlations between size and complexity across them, as a proxy for repetitive structure. For orientation, we compute both measures of position of the mark relative to the text, and also edge-based metrics. Several of these features are illustrated in Figure 3.1.

3.4.3 Technical Details

In this section, we give more of the technical details of our image processing algorithm. Readers not interested in image processing can skip this subsection, as these details are not strictly necessary for understanding the results. For specific features, see Appendix 3.9. The basic data representation of images is the raster array, which defines an image by an $h \times w$ grid of color values. The grid cells are called pixels, and the colors are typically broken down according to an underlying color model. The most common color model is the red-green-blue (RGB) system, which defines the full spectrum of colors by intensities on red, green, and blue color channels. Most image analysis algorithms are based on this representation of an image, and most data analysis software imports images in this form. An alternative representation, which we make use of in our own image processing algorithms, is the hue-saturation-value (HSV) color model, which is a cylindrical coordinates transformation of the RGB color space. It defines colors in terms of their hue, meaning the basic color itself, saturation, meaning how "intense" the color is, and

value, which refers to how bright the color is. Finally, greyscale images can be also represented through raster arrays as a single decimal value at each pixel, representing the intensity of light at that pixel.

Color Quantization through Density-based Clustering

The algorithm begins by learning how many distinct colors are in a given logo through a density-based clustering algorithm. Specifically, we employ the DBSCAN algorithm, which is a popular clustering algorithm which does not rely on a pre-specified number of clusters or distributional assumptions (Ester et al., 1996). Rather, it uses a density criterion to automatically determine both the number of clusters and cluster membership. DBSCAN is ideal for this application, as we know exactly the nature of the colorspace on which we are clustering, allowing us to specify a sensible density cutoff. Moreover, it is robust to noise.

We perform DBSCAN clustering on the HSV colorspace, which is a cylindrical coordinate transformation of the RGB colorspace that separates out the actual color value (hue) from other aspects of the color (saturation and lightness, also called value). Because of the cylindrical nature of the colorspace, hue (i.e. color) is represented along a circle, and hence the clustering must also operate over a circle, as shown in Figure 3.7. This is another benefit of DBSCAN: it does not rely on any assumptions about the distributions of the points or the geometry of the space, besides for being able to specify a suitable density metric.

A downside of DBSCAN is that it can be computationally inefficient, and the logos in our dataset can be quite large. Thus, we typically do DBSCAN on a random selection of pixels. Once we have identified the number of clusters through that, we use those same cluster centers in the standard k-means algorithm. The end

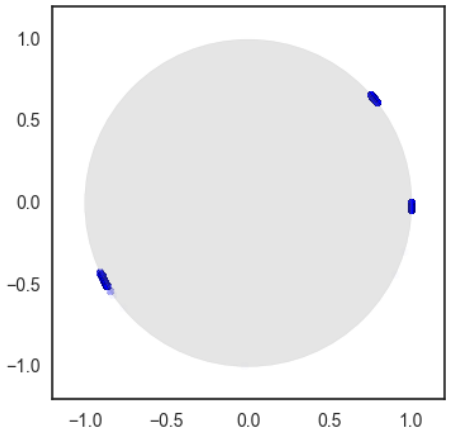

Figure 3.7: The three colors from Burger King's logo (blue, red, and yellow), plotted as the Hue value from HSV in polar coordinates. Here, red is the cluster of points at right, yellow is the cluster in the top-right, and blue is the cluster in the bottom-left. This is the space on which the DBSCAN clustering operates.

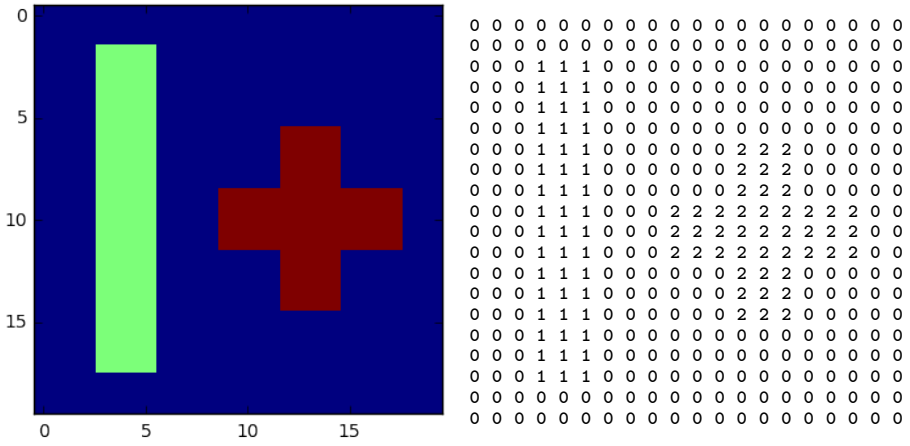

Figure 3.8: An example of color quantization: the image at left is quantized, yielding the matrix representation at right, where 0 corresponds to blue, 1 to red, and 2 to green.

result of the clustering is an assignment of each pixel in the original logo to a color cluster, or to the background. This is referred to as color quantization.

Region-based Segmentation

Computationally, quantizing the logo reduces the three dimensional raster array into a two dimensional matrix of cluster assignments. This is illustrated in Figure 3.8. Given this format, determining distinct regions of the logo is often as simple as

identifying connected regions of this matrix, and this, plus some steps to filter out noise and very small image segments, is how our algorithm proceeds. However, there are two complications. The first relates to text: in practice, some fonts are condensed to the point that two letters are slightly joined, leading the algorithm to think there is only one connected region, when there are in fact two distinct letters. The second complication relates to the mark, and is in some sense the inverse of the first: sometimes, a single mark may consist of several very closeby regions.

To address the first concern, we employ mathematical morphology, specifically the erosion and dilation operations. Erosion is a standard image processing technique that works on binarized images (background = 0, foreground = 1), transforming that image by assigning each pixel in the transformed image the minimum value within a pre-defined neighborhood of that pixel in the original binary image. Dilation is similar, but employing the maximum. In practice, what this means is that in erosion, connected regions are typically shrunk, whereas in dilation, they are expanded. To use these operations to help separate barely connected letters, we employ the following three steps: first, for every region isolated in the basic segmentation, we apply erosion, and identify any subregions generated by that erosion. Second, we separate those subregions, and then dilate them to approximately their original form. Finally, we run each of these new features through the font identification system defined in the next section. If any of them is identified as a font, the old region is discarded in favor of the subregions.

To address the second concern, we again apply DBSCAN clustering, this time using position on the logo as the quantity of interest. We set the density in the DBSCAN algorithm according to the size of the logo. This then finds mark pixels that are close together, regardless of whether or not they are actually connected.

Font Identification

For each of the segments identified through the above procedure, we first try to match them to a font. To do that, we standardize each segment to a grayscale 25×25 pixel representation, then apply template matching against our extensive collection of fonts, which have also been converted to the same representation. This representation is equivalent to representing each segment, and each font instance, as a length 625 vector, with values between 0 (black) and 1 (white). By template matching, we mean a simple distance calculation between the segment of interest, and each member of our font dictionary. In practice, this takes the form of a correlation between the entries in the segment vector and the entries in each font instance vector. We use a fairly simple heuristic to identify whether a segment represents a character: if the correlation between the segment and any font instance is greater than a certain cutoff, we say it is a match, and say that the segment matches the font with the highest correlation. We use different cutoffs, depending on the complexity of the segment, where complexity is measured by the perimetric complexity (the ratio of edge pixels to interior pixels). This is important because some letters, like i (which is represented without the dot), l, and o are very similar to commonly occurring mark features.

LAB Color Clustering

The colors within a given logo are represented in the continuous RGB color space. To convert these color triples to meaningful dictionary items, we then run another clustering algorithm on these triples across logos.[4] However, in order to cluster the

[4]The number of clusters both in this step and others was determined by the researcher, using scree plots.

colors, we need a sensible distance metric in this space. While RGB colors are the standard for computer representation, it is well established that distances in RGB color space do not correspond well to distances in human perceived distance. To rectify that, we employ another colorspace transformation, from RGB to the CIE-LAB (also just called LAB) colorspace, which is designed such that distances in colorspace correspond to differences in human perception of color (McLaren, 1976). Then we perform standard K-means clustering, resulting in the color dictionary shown in Figure 3.3.

Hull and Mark Clustering

To cluster both the hulls and the marks, we apply a similar procedure described above for fonts: we convert each hull and each mark to a 25×25 standardized greyscale representation, and then apply ordinary k-means clustering over the resultant length 625 vectors, determining the optimal number of clusters via scree plots. The only challenge is for the marks: the standardization procedure discards information about size. Yet, we also want to capture the different sizes of marks: a mark that forms the background of, and thus takes up 80% of a logo is different than one that takes up only 10%. To take this into account, we include an additional term in the clustering of marks, that adds weight to the fraction of the the logo's area taken up by the mark.

3.5 Descriptive Results

Now that we have a way of working with logos as data, we can start exploring the links between logo features and other aspects of the brand. Specifically, in this

section, we will look at links between three aspects of the brand: the firm's industry, as captured by Crunchbase, the firm's brand personality, from our MTurk surveys, and the logo. In all cases, we will seek to establish results from both descriptive statistics and visualizations, and by predictive models, in a unidirectional fashion, typically without considering the vast set of possible interactions between variables. Then, in the next section, we will build on this by building a model of logo design, that learns a joint representation across domains, which works with our full dataset, and takes such interactions into account.

3.5.1 Explaining Logo Variance

We first wanted to see whether or not the brand personality and the industry category of the brand explain anything about its logo. To do that, we considered all logo features as real-valued outcomes, and ran naive OLS regressions, saving the adjusted R-squared value from each.[5] We did this analysis in three separate batches: (1) predicting logos from industry, (2) predicting logos from brand personality, and (3) predicting logos from both together.

In Tables 3.1 and 3.2, we present the results for the most and least explained variance features, from regressions 1 and 2. In general, we find that brand personality scores capture much more variance than the industry codes, though this may also be attributed to the greater variance in the continuous brand personality scores, versus the binary industry labels. We find that features pertaining to the color palette tend to be the easiest to explain in both cases, including the mean and variance of the HSV colorspace's saturation and lightness (value) channels, the percentage whitespace,and a few of the color variables. Interestingly, in both cases,

[5]In many cases, the true variable is not real valued (see Appendix 3.9), but rather binary, and thus this approach will sometimes be underpowered.

Most			Least		
Feature	R^2	Adjusted	Feature	R^2	Adjusted
SD: sat	0.249	0.201	Dom. color: grey dark	0.037	-0.025
Mean: sat	0.187	0.135	Width: mixed	0.042	-0.02
Perc. white	0.164	0.111	Mark class: thin vertical rectangle	0.044	-0.017
GPC	0.137	0.081	Mark pos: absright	0.049	-0.012
Hor. symmetry	0.132	0.076	Mark class: wispy horizontal lines	0.049	-0.012
Color: yellow	0.13	0.074	Mark pos: top	0.053	-0.008
Font weight: bold	0.121	0.065	style mixed	0.054	-0.007
Hull type: rectangle-oval thin	0.12	0.063	Mark class: simple shapes	0.054	-0.006
Color: black	0.119	0.062	Mark class: long horizonal	0.054	-0.006
Down diagonals	0.118	0.062	Mark class: bad letters	0.054	-0.006

Table 3.1: The ten logo features with the most and least variance explained by brand personality, as captured by simple OLS.

Most			Least		
Feature	R^2	Adjusted	Feature	R^2	Adjusted
Hor. symmetry	0.147	0.084	Mark class: bad letters	0.025	-0.046
SD: sat	0.141	0.078	Dom. color: brown	0.038	-0.033
Mean: light	0.14	0.078	Dom. color: red dark	0.044	-0.026
Horizontal edges	0.135	0.072	Mark pos: bottom	0.045	-0.025
Perc. white	0.117	0.052	Mark pos: bot	0.047	-0.023
Entropy	0.114	0.049	Color: red dark	0.047	-0.022
Dom. color: blue medium	0.113	0.048	Mark class: bulky hollow geometric	0.048	-0.022
SD: light	0.111	0.046	Mark class: hollow circles	0.048	-0.022
Color: blue medium	0.111	0.046	Dom. color: blue dark	0.049	-0.021
Hull type: rectangle-oval thin	0.105	0.039	Mark class: long horizonal	0.052	-0.018

Table 3.2: The ten logo features with the most and least variance explained by industry codes, as captured by simple OLS.

Feature	Industry	BP	Both	Feature	Industry	BP	Both
SD: sat	0.078	0.206	0.222	down diag	0.022	0.069	0.081
Mean: sat	0.03	0.145	0.159	SD: light	0.046	0.052	0.077
Perc. white	0.052	0.123	0.157	Color: grey dark	0.008	0.051	0.073
Hor. symmetry	0.084	0.074	0.128	Color: black	0.024	0.051	0.065
Mean: light	0.078	0.055	0.104	Entropy	0.049	0.026	0.06
Horizontal edges	0.072	0.055	0.103	# Chars	0.011	0.052	0.057
Color: yellow	0.005	0.08	0.093	Color: red	0.038	0.045	0.056
GPC	0.019	0.081	0.09	# Colors	0.028	0.027	0.049
Hull type: rectangle-oval thin	0.039	0.062	0.083	ar	0.033	0.032	0.046
Font weight: bold	0.01	0.07	0.083	# Regions	0.016	0.052	0.046

Table 3.3: The 20 highest *adjusted* R^2 values from predicting logo features with both brand personality and industry codes, compared to the same adjusted R^2 from just the industry code model, and just the BP model. We see in almost all cases, a modest increase in adjusted R^2 from considering both sets of predictors jointly. Note that the number in the BP column may be slightly different than in Table 3.1, as several firms were missing industry codes, and had to be excluded.

the degree of horizontal symmetry also seems well explained, as do various aspects of complexity, including permetric complexity and entropy. The variables that are least explained by BP and industry tend to be those that either relate to the mark class, or those that tend to have very few observations associated with them, like logos with mixed font styles, or logos with the mark at the bottom.

In Table 3.3, we show what happens to the adjusted R-squared in regression 3, when we include both brand personality and industry codes in simple OLS to predict logo features. This illustrates the importance of jointly considering both what the firm *does*, as well as the firm's *brand identity*: in almost all cases, we find that the adjusted R-squared of including both sets of predictors is higher than either of the models in isolation. As this is adjusted for the number of predictors, this indicates that there is explanatory power by considering both sets of variables jointly.

3.5.2 Brand Personality Perceptions

In our data, brand personality provides an especially insightful portrait as to how consumers perceive the firm. To better understand the links between logo features and brand personality perceptions, we created a series of visualizations, sometimes referred to as forest plots. The goal of these visualizations is to understand the difference in consumers perceptions of brand personality for firms that have a certain logo feature, versus those that do not. We used these plots both as a way of validating our data by exploring some intuitive features like color and font, as well as a way of exploring the links between our data and prior literature. In all of the plots, we use the 15 brand personality factors described by Aaker (1997). We provide full access to all of the results, across three different factor structures of the

brand personality traits, through a web app.[6]

The first of these analyses is in Figure 3.9, where we compare consumer's brand personality perceptions across the three most common dominant logo colors: black, blue, and red. We can see, for instance, that black logos tend to score low on down-to-earth, but high on dimensions like daring, spirited, and imaginative. Interestingly, they also seem to score high on upper class and charming, but also on outdoorsy and tough. We will often see these variables moving opposite, except in this case. As we will see in other analyses, the likely cause here is that both types of logos do often feature black, but with vastly different stylings. Blue logos tend to be perceived as less daring, spirited, and imaginative, and also less upper class and charming, scoring high only on intelligent. Blue tends, generally, to be a fairly generic color, and thus it is not very surprising to find generally low responses with this feature. Finally, red scores very high on down-to-earth and wholesome, but low on upper-class. This is fairly consistent with our expectations, too: we often see logos of, for example, fast food, automotive, and hardware companies extensively featuring red.

In Figure 3.10, we see the same analysis for the accent color. What is interesting in this case is that the perceptions appear to move sometimes in the opposite way as in Figure 3.9. Black, for instance, appears to be a common accent color that does not exhibit much variation, but scores marginally higher in down-to-earth, which it scored low in when black was a dominant color. We see light green accent colors tend to score lower across the board, while yellow scores high on down-to-earth, but low on upper class, again reminiscent of, for instance, fast food logos. Finally, we see dark grey tends to imitate black.

[6] `https://rdew.shinyapps.io/bp_vis/`

In Figure 3.11, we explore different font features. In many cases, these features also match intuitions: serif fonts are perceived as more sophisticated, less rugged. Condensed lettering is more down-to-earth but less intelligent, while wide lettering is tough. Bold lettering and light lettering move in opposite directions, with bold letters being perceived as more down-to-earth and tough, while light letters are daring and sophisticated. Italics are tough and down-to-earth, but not upper class.

Finally, in Figure 3.12, we see some of the global descriptors. For instance, we find that high entropy logos, which tend to be complex, are perceived as more down-to-earth, but not daring, imaginative, intelligent, or upper class. Horizontally symmetric logos tend to be perceived better along almost all dimensions, except intelligent, perhaps reflecting the role of harmony in positive affect discussed in Henderson and Cote (1998). We find horizontal orientation is related to tough and outdoorsy brands, whereas upward-diagonal orientation appears positively related with cheerful, spirited firms. This latter point lends some support for the findings of Schlosser et al. (2016), who found that upward diagonals convey activity. Angularity, as captured by the number of corners, seems positively related to down-to-earth and tough logos, and negatively related to the others. This appears to support the findings of Jiang et al. (2015), where angularity is found to be associated with durability. Percentage whitespace's association with upper class and charming, and not with outdoorsy and tough is reminiscent of the findings of Semin and Palma (2014) about the femininity of white.

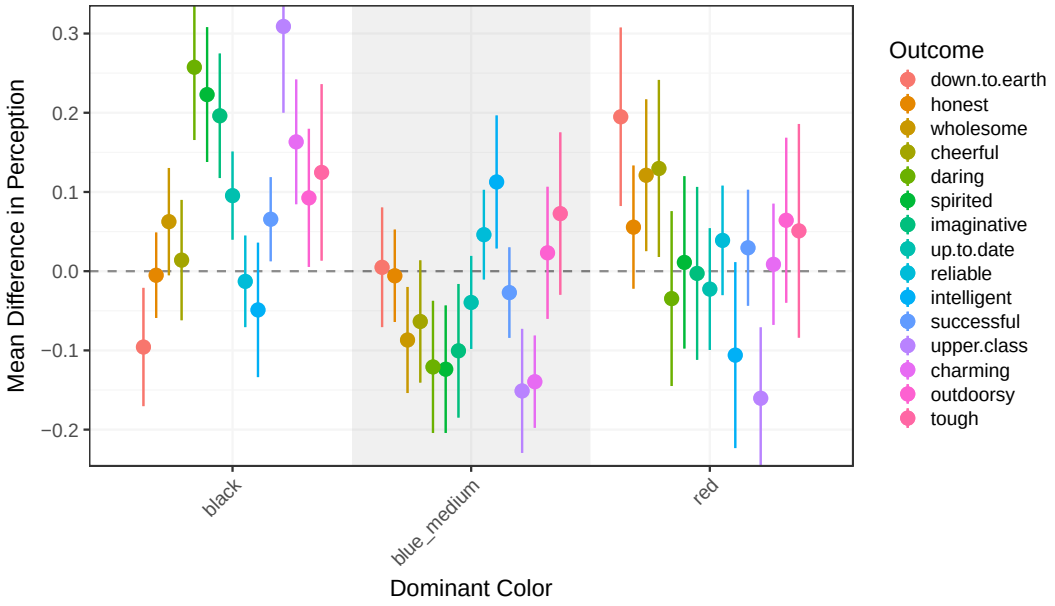

Figure 3.9: Forest plot for dominant color: each color in the plot represents a different brand personality factor, denoted in the legend. On the x-axis are the three most common dominant logo colors. On the y-axis, we see the mean difference in how consumers perceive logos with that dominant color, versus those without.

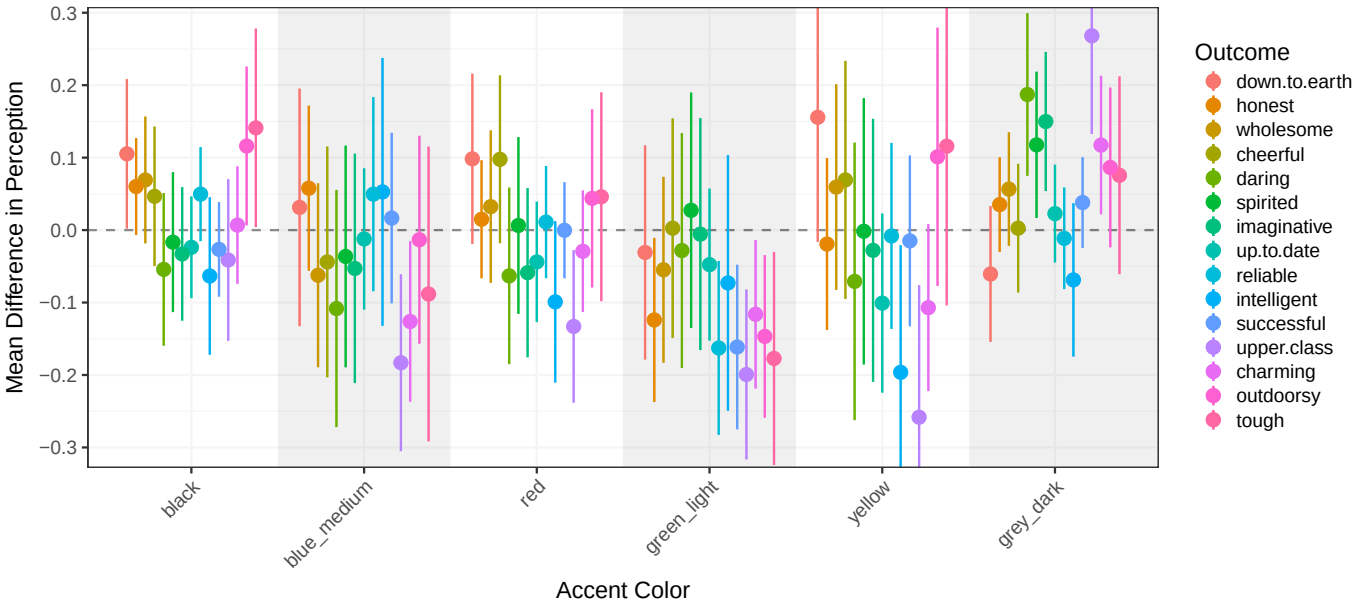

Figure 3.10: Forest plot for accent color: each color in the plot represents a different brand personality factor, denoted in the legend. On the x-axis are six accent colors. On the y-axis, we see the mean difference in how consumers perceive logos with that accent color, versus those without.

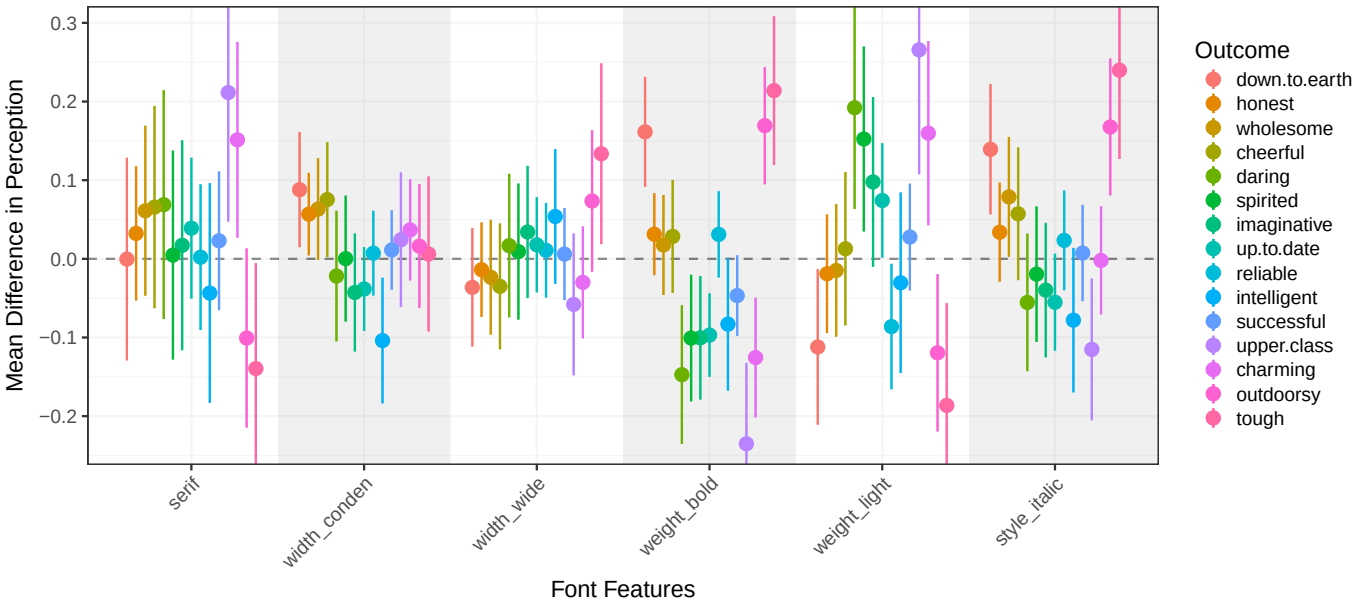

Figure 3.11: Forest plot for font features: each color in the plot represents a different brand personality factor, denoted in the legend. On the x-axis are six font features. On the y-axis, we see the mean difference in how consumers perceive logos with that font feature, versus those without.

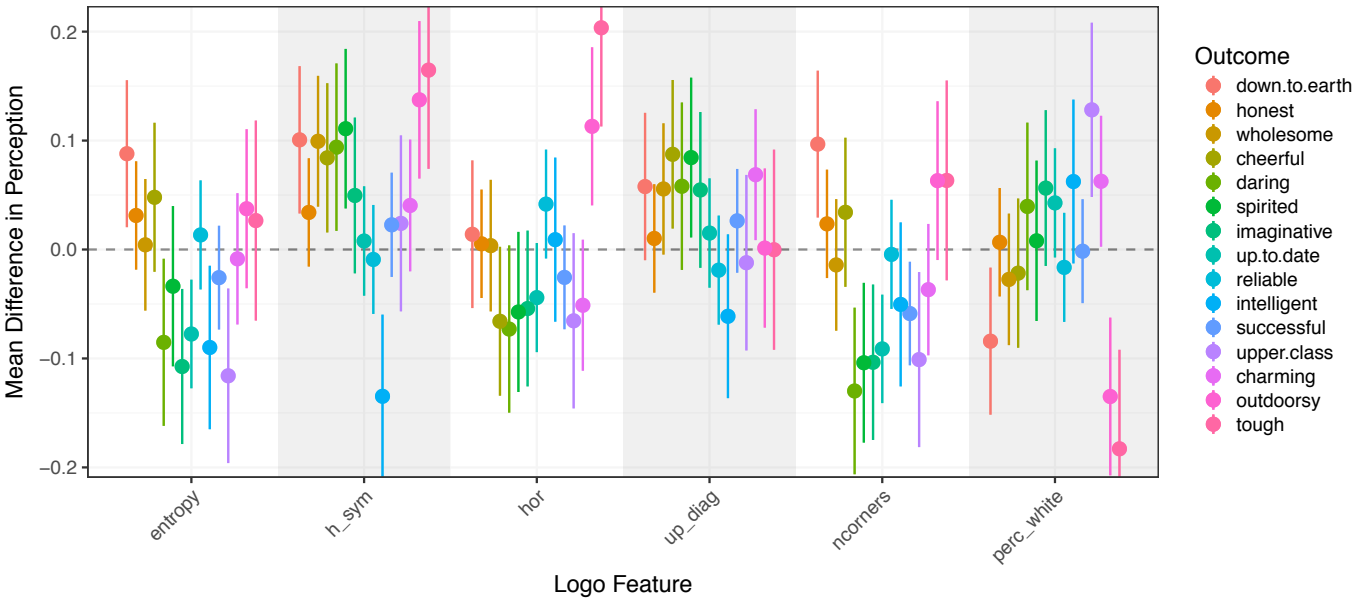

Figure 3.12: Forest plot for global features: each color in the plot represents a different brand personality factor, denoted in the legend. On the x-axis are six global features describing traits of the logo. To form these into binary variables, we used a median split. On the y-axis, we see the mean difference in how consumers perceive logos falling in the top half of logos, with respect to that feature, versus those in the bottom half.

3.5.3 Predictive Modeling

All of the previous analyses were simple, and largely descriptive in nature: in Section 3.5.1, we relied on R^2 from OLS, and in Section 3.5.2, we relied on mean differences. In all of these analyses, there may be huge multiple comparison problems: that is, we are looking for patterns across a huge feature space. It is entirely plausible that, in some cases, we may find correlations purley by chance. While it is helpful that, in almost all cases, these patterns match both our expectations and prior studies, we would still like to provide stronger evidence of these links between brands and their logos. In this section, we will attempt to do that by using regularized regression models, trained using cross-validation. We will estimate these models in two directions: predicting firm traits from logos, including firm industry and brand personality, and the reverse. This bidirectional analysis is meant to provide empirical support that each domain can be predictive of the others.

Specifically, we trained a variety of LASSO models using k-fold cross validation. Like OLS, LASSO attempts to minimize the squared error between the data and a linear prediction function. However, it also includes a penalization term that encourages sparsity of model coefficients. Specifically, for a continuous outcome like brand personality, the LASSO objective is given by:

$$\min_{\boldsymbol{\beta}} \frac{1}{N} \sum_{i=1}^{N} (y_i - \mathbf{x}_i' \boldsymbol{\beta})^2 + \lambda \sum_{p=1}^{P} |\mu_p|.$$

In all cases, we estimate an optimal λ through cross validation. The value of λ determines how many coefficients, μ_p, are nonzero. For a binary classification problem like the modeling of industry codes, the objective is changed to include the Bernoulli likelihood, rather than the mean squared error.

There are two primary insights we retain from these analyses: first, we look for the optimal penalization coefficient, λ, and how many nonzero coefficients the model optimally retains. Finding an optimal λ at which any coefficients are nonzero is a sign that the predictors do, in fact, have predictive power. Second, given there are any nonzero coefficients, we look at which predictors have nonzero coefficients, and which appear to be the strongest predictors.

Predicting Firm Traits from their Logos In Table 3.4, we report the results from the LASSOs of the brand personality factors on logo features. We see that some features are easier to predict from logo features than others, with honest, reliable, successful, and up-to-date being associated with the fewest significant predictors. For others, we find results similar to those described above. For instance, symmetric, red, bold logos predict a higher down-to-earth score. We see that the `sd_sat` variable appears frequently: `sd_sat` is zero for black and white logos, and higher for logos that feature contrasting, bold colors, like dark blue on a white background, or dark blue and bright red together. Hence, the strong negative association of `sd_sat` with upper class refers to the strong tendency of upper class logos to be black and white.

In Table 3.5, we repeat the analysis, but for the binary industry sector indicators. We see in this case that there were many industries for which the logo provided no signal. Omitted from the table were also many industry codes which were too sparse to use as a dependent variable. Among those that are there, we again find some intuitive patterns. For instance, clothing and apparel companies tend to be black and white, simple, with light font and few colors. Horizontal symmetry, sans-serif fonts, and red tend to indicate a firm is not in the financial services sector, while a blue and complicated are predictive that it is. Wide font is

BP Trait	# Coefs.	Top Predictors (Coef.)
Down-to-earth	43	Hor. symmetry (0.041), Color: red (0.034), Dom. color: dark red (0.033), Weight: bold (0.032), # Colors (0.029), SD: sat (0.029)
Honest	1	Dom. Color: yellow (0.003)
Wholesome	11	Hor. symmetry (0.029), # Chars (-0.02), Hull: rectangle-oval, thin (-0.015), Up diagonals (0.014), Color: black (0.013), Dom. Color: red (0.005)
Cheerful	21	Up diagonals (0.038), Hor. symmetry (0.032), mmark: avgright (-0.028), Dom. Color: blue: light (-0.025), Color: red (0.025), mark: class.circular (0.024)
Daring	20	SD: sat (-0.109), # Chars (-0.056), Hor. symmetry (0.023), # Corners (-0.021), Weight: bold (-0.018), entropy (-0.016)
Spirited	22	SD: sat (-0.065), # Chars (-0.062), Hor. symmetry (0.027), Up diagonals (0.022), Hull: rectangle-oval, large (0.022), entropy (-0.018)
Imaginative	9	SD: sat (-0.053), # Chars (-0.046), entropy (-0.04), # Corners (-0.018), mmark: avgbot (-0.009), mean: sat (-0.007)
Up-to-date	7	entropy (-0.031), # Corners (-0.029), mean: sat (-0.016), # Chars (-0.016), Weight: bold (-0.012), SD: sat (-0.007)
Reliable	6	Color: light green (-0.015), Dom. Color: grey: light (-0.009), down diagonals (-0.005), hor (0.001), Dom. Color: yellow (0.001), Weight: light (-0.001)
Intelligent	58	mean: sat (-0.056), Hor. symmetry (-0.051), # Corners (-0.051), SD: sat (0.044), Width: mixed (0.044), Width: condensed (-0.042)
Successful	7	Color: light green (-0.026), # Corners (-0.022), entropy (-0.012), SD: sat (-0.008), # Chars (-0.007), mark (-0.003)
Upper Class	34	SD: sat (-0.185), # Chars (-0.048), Weight: bold (-0.036), formserif (0.033), Hull: triangle (0.03), # Colors (-0.028)
Charming	15	SD: sat (-0.088), # Chars (-0.037), Up diagonals (0.025), mmark: avgleft (-0.018), formserif (0.018), gpc (0.016)
Outdoorsy	26	SD: light (0.047), Hor. symmetry (0.037), nregions (0.035), down diagonals (-0.032), Weight: bold (0.029), Hull: rectangle-oval, large (0.027)
Tough	25	SD: light (0.064), Hor. symmetry (0.046), nregions (0.044), down diagonals (-0.043), style: italic (0.042), % White (-0.038)

Table 3.4: LASSO results from predicting brand personality with logo features: we report, for each trait, how many nonzero predictors were found through cross validation on the penalization coefficient λ, and the most significant of those nonzero coefficients.

Industry Code	# Coefs.*	Top Predictors (Coef.)
Administrative Services	3	Dom. color: blue medium (0.09), Has mark (0.035), Dom. color: red (-0.003)
Biotechnology	5	# Colors (-0.265), Mark position: left (0.134), Mark class: horizontal complex (0.088), Font form: sans (0.022), Mark position: left (0.011)
Clothing and Apparel	16	SD: sat (-0.665), Entropy (-0.34), Font weight: light (0.331), # Colors (-0.26), Mark position: left (-0.244), Mark position: bot (0.129)
Commerce and Shopping	13	Color: green light (0.212), Dom. color: yellow (0.131), Color: blue light (-0.115), Color: red (0.112), Dom. color: blue light (-0.105), Font width: mixed (-0.086)
Community and Lifestyle:	20	Font width: mixed (-0.272), Dom. color: yellow (0.259), SD: sat (-0.254), Has mark (-0.239), Font style: mixed (-0.203), # Marks (0.196)
Consumer Goods	12	SD: sat (-0.399), Mark position: left (-0.225), Color: green light (0.177), Font style: mixed (0.104), Mark position: bot (0.102), Dom. color: brown (0.085)
Data and Analytics	8	Hull: rectangle-oval thin (0.302), Dom. color: grey dark (0.27), Hor. edges (-0.236), Mark class: thin vertical rectangle (0.218), Has mark (-0.153), AR (0.091)
Financial Services	24	Hor. symmetry (-0.295), Mark class: dense simple geometric (0.195), Dom. color: blue medium (0.184), Font form: sans (-0.183), Dom. color: red (-0.141), Entropy (0.125)
Food and Beverage	17	Color: green dark (0.264), Color: red (0.243), Color: brown (0.221), AR (-0.199), mean sat (0.192), Up diagonals (0.187)
Hardware	1	Entropy (-0.128)
Healthcare	9	Mark class: vertical narrow (0.156), Color: grey dark (-0.15), Color: blue medium (0.139), Mark position: left (0.107), Hor. symmetry (-0.085), perc white (0.031)
Information Technology	3	SD: light (-0.141), Mark class: very detailed (0.132), Dom. color: orange (0.005)
Internet Services	5	Has mark (-0.152), Font width: has condensed (-0.125), Color: yellow (0.108), Down diagonals (0.028), Mark class: wispy horizontal lines (-0.024)
Manufacturing	12	Font width: has wide (0.212), Font weight: light (-0.087), Mark class: detailed fit in circle (-0.071), Hor. symmetry (0.058), Dom. color: green dark (-0.056), Font weight: mixed (-0.055)
Media and Entertainment	3	Dom. color: grey light (0.183), Mark class: circular (0.083), Mark class: very detailed (0.036)
Software	1	# Chars (-0.059)
Telecommunications	1	Mark position: top (0.13)
Transportation	9	Hor. edges (0.301), Hor. symmetry (0.231), Color: grey light (0.127), Mean: light (-0.092), Mark class: very detailed (-0.039), Mark position: botFalse (-0.037)
Travel and Tourism	17	Up diagonals (0.378), Dom. color: brown (0.17), Mark class: circular (0.17), Color: black (-0.169), Mark class: bulky hollow geometric (0.16), Dom. color: red (0.141)

* The following industry codes could not be predicted better than chance (i.e. # Coefs. = 0): Consumer Electronics, Education, Energy, Government and Military, Lending and Investments, Mobile, Natural Resources, Payments, PlatFont form: s, Privacy and Security, Professional Services, Real Estate, Sales and Marketing, Sports, Sustainability

Table 3.5: LASSO results from predicting industry code with logo features: we report, for each trait, how many nonzero predictors were found through cross validation on the penalization coefficient λ, and the most significant of those nonzero coefficients.

predictive of manufacturing.

Predicting Logo Color from Firm Traits The other direction is also interesting: can we predict features of the logo, just by knowing the brand personality and industry of the firm? We showed in Section 3.5.1 some evidence indicating yes. Unlike brand personality and industry code, each aspect of the logo design has different statistical properties (e.g. real, binary, categorical, count), making it difficult to provide a unified predictive model of all, which is what we attempt to do in Section 3.6. In this section, we focus specifically on predicting the dominant color of the logo, as a case study of the broader logo prediction problem.

Again, we focus on LASSO regression, but this time using a categorical likelihood. We attempt to predict the dominant color, focusing on just six dominant color classes: black, blue, red, green, grey, and a combination of yellow-orange-brown. In all cases, yellow-orange-brown could be predicted no better than chance. This was by far the smallest of the classes, and we omit it from all results.

Color	# Coefs.	Top Predictors (Coef.)
black	7	glamorous (0.327), family oriented (-0.252), tough (0.107), spirited (0.105), rugged (0.076), hard working (0.047)
blue	9	intelligent (0.2), good looking (-0.18), unique (-0.143), corporate (0.134), daring (0.083), smooth (-0.045)
green	2	original (-0.201), upper class (-0.018)
grey	3	down-to-earth (-0.21), independent (-0.054), upper class (0.015)
red	6	upper class (-0.308), leader (0.077), down-to-earth (0.071), western (0.024), successful (0.016), reliable (0.015)

Table 3.6: LASSO results predicting the categorical outcome dominant color from brand personality.

Color	# Coefs.	Top Predictors (Coef.)
black	4	Clothing and Apparel (0.136), Consumer Goods (0.041), Apps (0.021), Biotechnology (-0.006)
blue	7	Financial Services (0.253), Health Care (0.193), Administrative Services (0.103), Energy (0.065), Sustainability (0.034), Manufacturing (0.002)
green	0	NA
grey	0	NA
red	2	Food and Beverage (0.212), Administrative Services (-0.011)

Table 3.7: LASSO results predicting the categorical outcome dominant color from the binary industry codes.

Color	# Coefs.	Top Predictors (Coef.)
black	10	glamorous (0.323), family-oriented (-0.104), rugged (0.091), unique (0.088), Apps (0.023), spirited (0.008)
blue	10	Financial Services (0.209), Health Care (0.166), good looking (-0.13), Administrative Services (0.094), corporate (0.085), Sustainability (0.04)
green	0	NA
grey	1	Data and Analytics (0.003)
red	5	Food and Beverage (0.159), down-to-earth (0.094), upper class (-0.069), Administrative Services (-0.038), family-oriented (0.02)

Table 3.8: LASSO results predicting the categorical outcome dominant color from both brand personality and industry codes. We see that both play a role in determining the outcomes.

What is interesting to note in this analysis is not just what is present when each set of predictors is used in isolation, but how the optimal set of predictors changes when we go to the full model. First, we note that there are both sector predictors, and personality predictors present in the optimal set of predictors. However, we also see, for instance, that the sector Food and Beverage seems to have replaced many of the personality signals in predicting a red dominant color, although for black, personality signals remain dominant. Blue is also largely

predicted by sector, although the presence of the brand personality trait "good looking" as a negative predictor provides further evidence of the commonness of that color.

3.5.4 Building Personality-Consistent Logos

As a final analysis before proceeding to our model of logo design, in this section, we use the descriptive links we established between a brand's personality and its logo to automatically build a logo template consistent with a given brand personality profile. To do that, we selected a set of easy to use logo features, then built simple generalized linear models to predict those outcomes, based solely on the original 42 brand personality traits. We then designed an *R* program that algorithmically translates the model predictions into a set of pre-defined features matching those predictions, which are then unified into a logo. Those selected features, together with their models and engineered features, are described below:

- Dominant color: multinomial logit. The color with the highest probability is the color used by the program to color the text and, if there is no accent color, the mark.

- Accent color: logistic regression combined with a multinomial logit. We first model whether or not there is likely to be more than one color, then model what that color is. The highest probability accent color is what determines the color of the mark.[7]

[7]We do not exclude the dominant color as a possible accent color in the model. However, in generating the template, if the most likely accent color is the same as the dominant color, we use the second most likely accent color.

- Font family: multinomial logit. We chose five representative font families from our font dictionary. The family with the highest predicted probability is used to display the logotype.[8]

- Number of mark regions: multinomial logit. Rather than model this as a count, we modeled this as a choice over five possible classes, corresponding to between zero and four mark regions. This helped constrain what the algorithm could generate. The highest probability class determines whether there is a mark, and if so, how many subregions it has, which is a proxy for complexity.

- Angularity: multinomial logit. We model angularity as a choice between four classes, corresponding to increasing number of corners, generated by bucketing the original `ncorners` variable by quartiles. In the logo generation procedure, the highest probability class generates shapes with more corners, ranging from a circle (lowest), to a complex star shape (highest).

- Percentage whitespace: logistic regression. We predict the actual percentage whitespace, then translate this into a variable that increases the line width of the mark, or, past a certain threshold, fills in the mark. This also controls the boldness of the font.

- Horizontal orientation: multinomial logit. We bucket the original `hor` variable into three buckets, low, medium, and high horizontal orientation. For medium horizontal orientation, we add one horizontal line to the logo, under the logotype. For high horizontal orientation, we add lines both below and above the logotype.

[8] In some cases, due to copyrights on certain fonts, we had to rely on a close proxy in generating the logo.

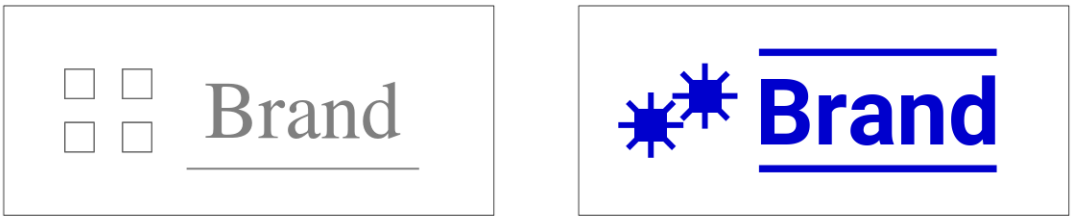

Figure 3.13: Two logos generated from the app. At left, we generated a logo that is high in sophistication, and low in all other dimensions, using the five brand personality superfactors. At right, we generated a logo that is high in ruggedness, competence, and sincerity, but low in sophistication and excitement.

We give two examples of generated logos in Figure 3.13. These logos were intentionally generated to show the range of features. The final product of this modeling and feature engineering procedure can be accessed and experimented with by the reader online at: `https://rdew.shinyapps.io/logo_gen/`.

3.6 Model of Logo Design

In this section, we describe our model of logo design. Specifically, our model draws on methods from deep generative modeling (Rezende et al., 2014; Kingma and Welling, 2013; Ranganath et al., 2014b) and multiview learning (Li et al., 2016) to learn joint representations of brands that can then used to predict each of our domains of interest: how the firm describes itself in text, what features are in the firm's logo, and how consumers evaluate the brand's personality.

Our modeling framework is based on the variational autoencoder (henceforth, VAE), a deep learning model designed to learn generative models of data (Rezende et al., 2014; Kingma and Welling, 2013). Conceptually, given data that contains many instances of some phenomenon of interest (e.g. a collection of hand-written

digits, or in our case, a collection of brand identities), a VAE learns to recreate that data, by representing the instances as low dimensional latent representations, sometimes called the latent code. A VAE contains two components: an inference network, and a decoder network. The inference network is a deep neural network that takes data as its input, and outputs the parameters of an approximate posterior distribution for the latent representations of the data it was given. The decoder network is a separate deep neural network that takes these latent representations of the data, and outputs a probability distribution over the original data. In this way, the two together provide dimensionality reduction: the inference network doing the inference from data, distilling the data into a low dimensional vector representation, and the decoder making predictions from that representation.

Our specific implementation is a multimodal variational autoencoder (MVA), which learns a representation of the joint distribution across many domains of interest. Similar models have been studied in a number of recent papers, including Ngiam et al. (2011); Feng et al. (2014); Wang et al. (2015); Suzuki et al. (2017); Vedantam et al. (2018), and Wu and Goodman (2018). In generality, suppose we have D domains of interest, indexed by $d = 1, \ldots, D$. In our application, these will be text, logo, and brand personality. For each of the brands in our data, $b = 1, \ldots, B$, denote the data of brand b in domain d as x_b^d. In each domain, there are different features (words for text, logo features for logos, personality traits for brand personality). We will index these features $j = 1, \ldots, V_d$. Note that, in our work, we observe data in all of the domains for each brand, but that the framework also allows for missingness.

3.6.1 Generative Model

The heart of our model is learning a multimodal representation of brands. Specifically, we assume that the full set of features of a given brand b across all domains can be distilled into a vector, z_b, of length K. Substantively, z_b can be thought of as brand b's brand identity, as this representation will be used to predict which features will be used in describing the brand, and how people view the brand. To learn this latent representation, we model each component of this representation, z_{bk}, with a unit normal prior, in keeping with the standard VAE:

$$z_{bk} \sim \mathcal{N}(0, 1).$$

Conditional on z_b, we assume each of the domains are independent, such that the joint distribution of data and latent variables factorizes as:

$$p(x_b^1, \ldots, x_b^D, z_b) = p(z_b)p(x_b|z_b) = q(z_b)\prod_{d=1}^{D} p(x_b^d|z_b). \tag{3.1}$$

Moreover, we assume that each of these conditional distributions, $p(x_b^d|z_b)$, factorizes into feature-specific probability models,

$$p(x_b^d|z_b) = \prod_{j=1}^{J_d} p(x_{bj}^d; \mu_{bj}^d = f_{dj}(z_b)), \tag{3.2}$$

where the parameters of these models are functions of the latent representation. We will also use the generic notation, $x_{bj}^d \sim p_d(x_{bj}^d; \mu_{bj}^d)$, to denote these models. Examples include a normal distribution for real-valued data (e.g. entropy), a Bernoulli distribution for binary data (e.g. words), and a categorical distribution for choice data (e.g. colors). We elaborate more on the specific models used in our work in the next section.

As shown in Equation 3.2, the parameters μ_{bj}^d are computed from the latent representation z_b through a domain-specific function, $(f_{d1}(z_b), \ldots, f_{dJ_d}(z_b) = f_d(z_b)$. We will assume that this function is a deep neural network, akin to the joint multimodal variational autoencoder in Suzuki et al. (2017). In the VAE literature, this function is often referred to as the decoder network, as it "decodes" the latent representation into predictions about the data. In specifying these networks, we will use dense layers with rectified linear activation units (ReLU) and skip connections, which means the following sequence of computations:

$$
\begin{aligned}
\mathbf{h}_{b1}^d &= \max(0, \mathbf{a}_{d0} + W_{d0} z_b) \\
&\ldots \\
\mathbf{h}_{bL_d}^d &= \max(0, \mathbf{a}_{d(L_d-1)} + W_{d(L_d-1)}^h \mathbf{h}_{b(L_d-1)}^d + W_{d(L_d-1)}^z z_b) \\
\mu_{bj}^d &= a_{dL_dj} + (\mathbf{w}_{dL_dj}^h)'(\mathbf{h}_{bL_d}^d) + (\mathbf{w}_{dL_dj}^z)' z_b
\end{aligned}
$$

Intuitively, this set of equations sequentially applies the same operation, called the rectified linear unit, or ReLU. At each layer of the model, the ReLU computes a new representation of the brand, which we call the hidden units at layer ℓ, denoted by $\mathbf{h}_{b\ell}^d$. We combine these hidden units with the original representation z_b, in what is known as a skip connection, to learn the hidden units of the next layer. This operation is repeated L_d times for the number of layers in the network for domain d. At each layer, the number of hidden units (meaning the dimension of $\mathbf{h}$) may change, which allows the network to learn different levels of abstraction of the data. Moreover, as the operations are nonlinear, this network theoretically corresponds to learning an arbitrary nonlinear relationship between the data and the representation. In effect, this means we can capture quite complex joint distributions across the features. The more hidden units, and the more layers, the more expressive the model. We include skip connections to avoid a phenomenon

called latent variable collapse, in which models like ours get stuck in uninformative local optima (Dieng et al., 2018). We denote the whole set of parameters,

$$\theta_{dj} = (a_{dL_dj}, w_{dL_dj}, \{a_{d\ell}, W_{d\ell}\}_{\ell=1,\ldots,L_d-1}),$$

and this whole operation as:

$$\mu_{bj}^d = \text{DNet}_d(z_b; \theta_{dj}),$$

where DNet$(\cdot)$ stands for "decoder network." Note that in any given domain, across the features j, many of the components of θ_{dj} will be shared. We may also use θ_d to refer to all of the network parameters within domain d across all j. We describe the specifics of each domain's network in a later section.

3.6.2 Domain Probability Models

Conditional on the joint representation z_b, each brand's features are modeled using domain-specific probability models, which factorize across features j, and the parameters of which are inferred from the decoder network. The specific models used for our data are:

- *Text:* For determining which words to include, we stemmed and tokenized the full vocabulary, removed standard stopwords, then filtered out words that occurred in less than twenty different brand descriptions. For modeling this textual data, we then use a simple binary model, capturing whether or not a given word is present in the textual description. That is, for each brand b, for

each word w, we model:

$$P(x_{bw}^{\text{Text}} > 0) = \frac{1}{1 + \exp(-\mu_{bw}^{\text{Text}})},\ \mu_{bw}^{\text{Text}} = \text{DNet}_{\text{Text}}(z_b; \theta_{\text{Text},j}). \tag{3.3}$$

This simple coding captures the idea that firms choose to use a set of words, and that we are interested in whether or not a firm chooses to label itself a certain way (e.g. as "innovative"). Although the number of times a given word is repeated may contain information, it may also merely reflect how much text was present on the firm's website, or any number of unrelated factors. Hence, we only model whether or not a given word is present.

- *Logo features:* Many of the logo features exhibit very different statistical properties. In the appendix, we describe all of the logo features, together with their data types. In our model, conditional on the logo-specific parameters θ_{Logo} and the latent representation z_b, each of these features is drawn independently. For each one of these features, we then use an exponential family distribution that has support on that data type. Specifically, for real-valued data, like entropy, we use a normal distribution (or a lognormal distribution for continuous values with only positive support), such that for a real-valued feature indexed j, we have:[9]

$$x_{bj}^{\text{Logo}} \sim \mathcal{N}(\mu_{bj}^{\text{Logo}}, \sigma_{bj}^{\text{Logo}}),\ (\mu_{bj}^{\text{Logo}}, \log(e^{\sigma_{bj}^{\text{Logo}}} - 1)) = \text{DNet}(z_b; \theta_{\text{Logo},j}) \tag{3.4}$$

Note that, for two parameter families, like the normal, we learn both the mean and the variance. For binary data, like whether the logo has a mark, we use a bernoulli distribution, equivalent to the model for text described above.

[9]The $\log(e^y - 1)$ structure in Equation 3.4 is the inverse of the so-called softplus function, $y = \log(1+e^x)$, which is commonly used to enforce positivity, as a more numerically stable alternative to a simple exponentiation.

For choice data, like the dominant color, where we have one of $m = 1, \ldots, M_j$ possible options, we use a categorical distribution, such that:

$$x_{bj}^{\text{Logo}} \sim \text{Categorical}(\text{Softmax}(\boldsymbol{\mu}_{bj})), \tag{3.5}$$

$$\boldsymbol{\mu}_{bj} = (\mu_{bj1}, \ldots, \mu_{bjM_j}), \tag{3.6}$$

$$\mu_{bjm} = \text{DNet}(z_b; \theta_{bjm}). \tag{3.7}$$

- *Brand personality:* Similar to the real-valued logo features, brand personality in our data is also real-valued: it is the average of all respondents ratings, measured between 0-4. We approximate this using a normal model, again with the mean and variance learned from the latent representation.

3.6.3 Inference

The key task in using the MVA framework is learning the representations z_b. Once we know z_b, we can use z_b to make predictions across modalities via the probabilistic decoder. Important to our framework, we would like to be able to learn z_b given information on only a subset of the domains. Then, we can use the representation z_b and the decoder to make predictions for the unseen modalities. In practice, this means we could use the MVA to generate a logo template, given a textual description, to generate words describing a specific set of logo features, or to predict brand personality assessments given either visual or textual information.

In the standard VAE, the inference network is a map from the data x_b to parameters of a variational approximation to the posterior of the latent representation z_b. In most models, learning latent parameters is accomplished by model training, using either maximum likelihood, MCMC, or variational inference.

Inference networks transform the problem of inference of latent parameters into a problem of learning a function, parametrized by a neural network, such that given any data, we can obtain an approximate posterior distribution for the latent variables of interest, simply by evaluating the function. Using similar notation as above, a generic inference network can be written as:

$$\xi_b = (\xi_{b1}, \ldots, \xi_{bK}) = \text{INet}(x_b; \phi),$$

where ξ_{bk} is the vector of parameters of a (mean field) approximation to the true posterior, $q(z_{bk}; \xi_{bk}) \approx p(z_{bk}|x_b)$. In the case of a VAE, this approximation is assumed to be normally distributed, such that:

$$q(z_{bk}; \xi_{bk}) = \mathcal{N}(z_{bk}; \mu = \xi_{bk1}, \sigma = \xi_{bk2}). \tag{3.8}$$

We will denote this inference network parametrized variational distribution by $q_\phi(z_b; x_b)$.

Inference in VAEs is typically done using Variational EM (VEM), as introduced in Rezende et al. (2014) and Kingma and Welling (2013). In the case of the standard VAE, where there is just one decoder network, and an inference network as given above, the following loss function is minimized:

$$\ell(\theta, \phi) = \sum_{b=1}^{B} -E_{q_\phi(z;x_b)} \left[\log p_\theta(x_b \,|\, z)\right] + \text{KL}(q_\phi(z; x_b) \;||\; p(z)). \tag{3.9}$$

This loss is exactly the (negative) evidence lower bound (ELBO) for doing variational inference on the latent parameters, z, but where the variational approximation is given by the inference network (Blei et al., 2017). This procedure is referred to as variational EM, as the variational distribution approximates the distribution of the latent variables z, and the model parameters θ are optimized for

the likelihood of the data. Another interpretation of this loss is that the first term encourages a good reconstruction of the data, while the second term regularizes estimates toward the prior.

To estimate the MVA, we make three modifications to the standard inference algorithm. The first modification is to allow for the multiple modality-specific decoder networks. This adjustment is straightforward: as shown in Equation 3.1, we assume $p_\theta(x_b \mid z) = \prod_{d=1}^{D} p(x_b^d \mid z)$. Hence, the ELBO above decomposes into a series of domain-specific terms:[10]

$$\ell(\theta, \phi) = \sum_{b=1}^{B} -E_{q_\phi(z;x_b)} \left[\sum_{d=1}^{D} \log p_{\theta_d}(x_b^d \mid z) \right] + \mathrm{KL}(q_\phi(z; x_b) \mid\mid p(z)). \tag{3.10}$$

Since we assumed the joint factorizes in Equation 3.1, marginalizing over unobserved modalities is trivial: we can simply ignore them, computing the sum in 3.10 for only those domains which are present.

The second modification is to allow for inference given only a subset of the modalities. Prior research has worked to address this problem in classical autoencoders in Ngiam et al. (2011), and in variational autoencoders in Wang et al. (2015); Suzuki et al. (2017); Vedantam et al. (2018) and Wu and Goodman (2018). Mathematically, given information on only a subset of the modalities, $\tilde{x}_b$, we want to approximate the posterior $q_\phi(z_b|\tilde{x}_b) \approx p(z_b|\tilde{x}_b)$. Given such approximations, the modification to the ELBO is straightforward:

$$\ell(\theta, \phi) = \sum_{b=1}^{B} -E_{q_\phi(z;\tilde{x}_b)} \left[\sum_{d=1}^{D} \log p_{\theta_d}(x_b^d \mid z) \right] + \mathrm{KL}(q_\phi(z; \tilde{x}_b) \mid\mid p(z)). \tag{3.11}$$

[10] A similar decomposition is used in, for example, Wu and Goodman (2018), where they also weigh each term of the decomposition by a factor λ_d. In this work, we choose to give each term equal weighting, though adjusting the weights may provide a solution or means of understanding how the MVA trades off each domain in computing the posterior

However, as described in Wu and Goodman (2018), using this loss assumes we can approximate $p(z_b|\tilde{x}_b)$ for any combination of missing modalities. In the general case, this entails learning 2^D different approximations, which is equivalent to learning 2^D inference networks, one for every possible combination of present modalities. In their work, Wu and Goodman (2018) propose a mixture of experts approach to deal with that exponential growth in computation. In our work, we simplify the problem by restricting the patterns of missingness we allow. Specifically, we assume that we are given either the full data, or only one of the modalities. This approach entails only learning $D + 1$ inference networks. Mathematically, we denote the inference network associated with the data from domain d as $\text{INet}_d(x_b^d; \phi_d)$, and the full data inference network by $\text{INet}_{\text{Full}}(x_b; \phi_{\text{Full}})$. We again simplify notation by letting $\phi = (\phi_{\text{Full}}, \phi_1, \ldots, \phi_D)$. Intuitively, the output of the domain-specific networks is the model's "best guess" of the posterior distribution, given data from only one domain.

The final modification is to the actual training procedure. Inspired by the (non-variational) approach in Ngiam et al. (2011), during training, we randomly hold out certain modalities, and force the model to reconstruct those modalities.[11] This forces the model to learn multimodal representations, and avoids the case wherein some of the latent dimensions specialize in predicting only one domain. More specifically, in optimizing the loss in Equation 3.11, at each iteration of the optimization, the data is randomly split into $D + 1$ bins. Then the model is trained as if those brands had missing data, using the full and modality-specific inference networks. At the next iteration, the brands are reshuffled randomly among the bins. At any point during training, each brand is only present in one of the bins. Thus, this shuffling procedure can also be thought of as minibatch inference for each of the inference networks, but where the minibatches are non-overlapping across bins. By

[11]This procedure is similar in spirit to the denoising autoencoder of Vincent et al. (2008).

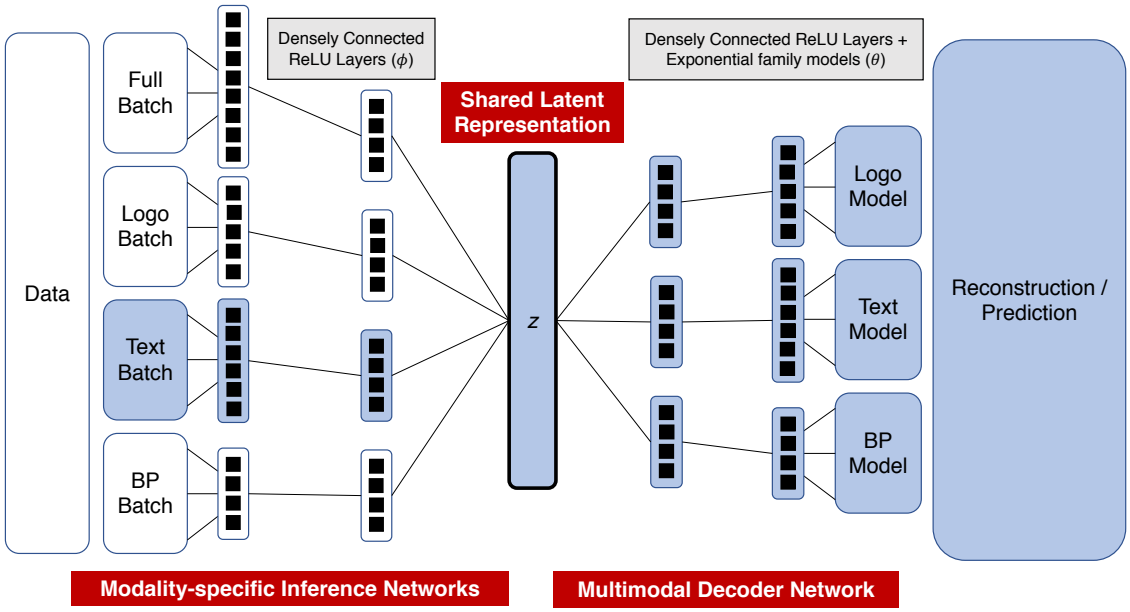

Figure 3.14: An illustration of the MVA framework: at left, the full data is subdivided during training intervals into both full and heldout batches. For each of these, an inference network is learned to approximate the posterior of the latent representation, z_b. From these representations, the data is reconstructed through decoder networks, where domains are assumed to be conditionally independent, given z_b, and where the predictions come through feature-specific exponential family models. This framework enables transfer learning acorss modalities, as illustrated by the blue shading: given, for example, text describing a brand, the model can then infer an approximate posterior distribution for the representation z_b, which can then be used to predict all modalities via the decoder networks.

restricting the procedure to non-overlapping bins, we avoid the problem of having to weigh the inferences across multiple inference networks.

Intuitively, this shuffling procedure works because what is being learned during training is not z_b, but two sets of functions—the decoder and inference networks—which are defined by global parameters, θ and ϕ. By randomly shuffling observations across the modalities, these parameters learn to construct representations for a greater variety of data inputs than if the data were not reshuffled throughout. That is, for any given brand, the model (in expectation) learns to reconstruct all its features from each modality, as well as from the full set of features. We give a schematic representation of the full model structure in Figure 3.14. Ultimately, what this framework and training procedure enables is transfer

learning: that is, it gives us the ability to use modality-specific inference networks to learn a representation z_b that is predictive of all of the modalities, even if only one of those modalities is present at test time.

3.6.4 Network Structures and Estimation Details

The final ingredient in deploying our MVA framework is specifying the structure of the decoder and inference networks. As described previously, we use a skip structure in our decoder network, as described by Dieng et al. (2018). This structure avoids a phenomenon called latent variable collapse, in which the model learns uninformative representations of the data very close to the prior. The remaining structure of the data is then specifying the dimensionality of the latent representation z_b, as well as how many layers, and how many hidden units are in each layer, for each domain of the data, for both the decoder and inference networks.

In this work, both our layer sizes, and the number of layers employed are small, relative to much of the deep learning literature. This is because we are using already somewhat structured and pre-processed inputs, that are already represented at higher levels of abstraction. Specifically, we assume there are 10 latent variables ($K = 10$). In all of the decoder networks, we assume two layers, with 20 hidden units in the top layer for each. For the logo and brand personality networks, we use a bottom layer of 40 units. For text, we use 60 hidden units in the bottom layer, reflecting its higher dimensionality than other domains. We employ batch normalization between the top and bottom layers, which we have found greatly improves the learned models (Ioffe and Szegedy, 2015).

Under our MVA framework, we assume a four part structure for the inference network: one inference network for each of the domains, and a fourth

inference network which is given access to all of the domains. For these, we assume an asymmetric structure across domains, reflecting our assumptions about how information rich each domain is. Across all of the inference networks, we assume that the topmost layer has 20 hidden units. For the full information inference network, we assume a bottom layer with 80 hidden units. For the text, logo, and BP inference networks, we assume a bottom layer with 40 units.

We implement the model estimation using the Edward probabilistic programming language (Tran et al., 2016), which is built on Tensorflow. Edward facilitates inference for deep, probabilistic models, by leveraging black box variational inference with stochastic gradients (Ranganath et al., 2014a), and automatically incorporating procedures like the reparametrization trick (Kingma and Welling, 2013) to facilitate estimation. We optimize the model using Tensorflow's implementation of the Adam optimizer with stepsize 0.001, using single sample stochastic gradients.

3.7 Model Results

3.7.1 Model Fit

Reconstruction Error

The metric by which VAEs are often evaluated is what's called reconstruction error: how well does the model do at reconstructing the data it is meant to represent? In our case, for each inference network, the error can be decomposed into the part that is own-modal reconstruction error (the modality that was input to the network), and a part that is cross-modal reconstrucion (i.e. the heldout modalities). In Table

Feature(s)	Full Data	Logos	Text	BP	Intercept Only
Binary Text	0.096	0.102	0.094	0.126	0.157
Binary Logo	0.122	0.135	0.126	0.182	0.212
Real Logo	0.472	0.504	0.487	0.686	0.753
BP Ratings	0.190	0.200	0.181	0.210	0.405

Table 3.9: The own- and cross-modal reconstruction error across all of the inference networks, relative to an intercept only model.

3.9 we compare absolute error rates across the inference networks for several components of the model, using the last batch of training as the input data for the inference networks. We compare this to an "intercept-only" benchmark, wherein the average value of each feature is used as the prediction for all inputs.

There are three interesting patterns to note: first, the model is able to reproduce the data significantly better than a naive intercept-only model. Second, we notice that the BP-based inference network does worse on all cross-modal reconstruction errors. This is not surprising: relative to the other modalities, brand personality is a very high-level, abstract input, with significantly fewer features. As such, it is unable to match the representations learned by the other inference networks. Finally, for all networks except brand personality, we find that the reconstruction error rates are roughly equivalent. This is because, in all cases, the decoder network is the same, regardless of the inference network, and moreover, at each iteration, the firms that are used in each inference network are randomly shuffled. Hence, the model is incentivized to learn coherent representations across the inference networks, which result in nearly equivalent hit rates.

Data Complementarity

Given these patterns in the reconstruction error, we are also interested in understanding to what degree the learned representations are coherent across

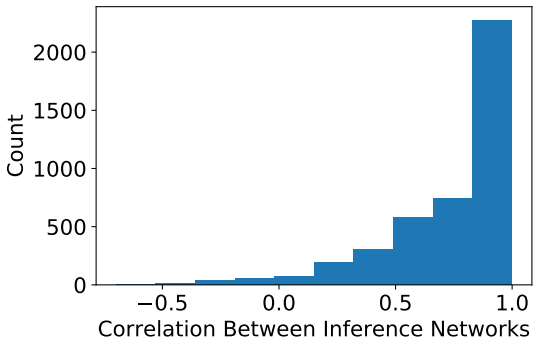

Figure 3.15: Histogram of the correlation of the expected value of z_b as inferred across inference networks, illustrating the high agreement across the different networks.

inference networks. In general, and as implied by the hit rates above, we find that the representations learned are coherent across inference networks. More specifically, in Figure 3.15, we plot the histogram of within brand correlations of z_b across inference networks. That is, to what degree is the posterior mean of z_b when inferred through, for example, the full inference network, correlated with, for example, the brand personality inference network? We see in Figure 3.15 that the histogram is hugely skewed toward 1, implying a large degree of correlation across inference networks. The left tail, howeover, implies there are some disagreements.

In Table 3.10, we show the average correlation between representations, averaging over all brands. From this, we see again that there is by and large agreement, but that by using just brand personality, we get correlated, but not equivalent representations. This again explains why the hit rates vary the way they do: brand personality is not rich enough to achieve the same degree of precision as the other modalities. We can also see this same pattern in Figure 3.16, where we plot the representations across inference networks against each other: we see again that all are correlated, but that the strength of the correlation varies. The full and text networks learn the closest representations, which makes sense, given the richness of those two data sources. Brand personality has a lower correlation.

	Logos	Text	BP
Full Data	0.9	0.966	0.575
Logos	.	0.899	0.568
Text	.	.	0.576

Table 3.10: Average over brands of the correlation between z_b as learned by different inference networks.

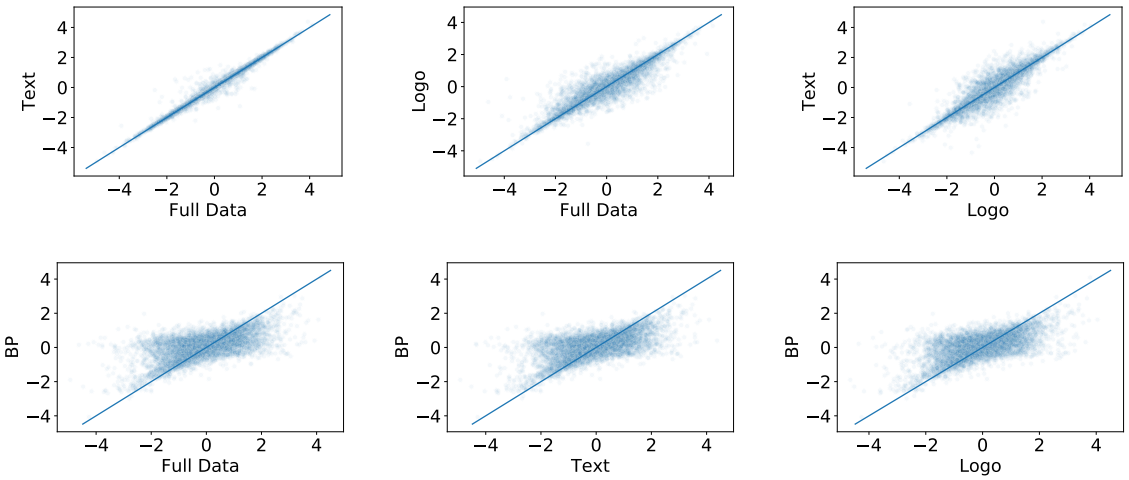

Figure 3.16: Scatterplots showing how the z_b learned from one inference network compares to the z_b learned from another, where all components of z_b are collapsed and plotted jointly. The line is the 45 degree line, illustrating perfect agreement between representations.

Interestingly, the brand personality is strongly correlated with the full representation in several of the dimensions, and weakly correlated in others. For instance, the average correlation of z_{b4} as learned by the text data with that of brand personality is 0.9, while the average correlation of z_{b7} is a mere 0.1. This supports the idea that these modalities are complementary: brand personality captures some aspect of the brand that's displayed in text and logo, but that other aspects of textual and visual identity are independent of the brand personality.

3.7.2 Exploring the Latent Space

Neighbors in z Space

Now that we understand how the model approximates the multimodal representation, we can start exploring what representations it learns. In general, it is difficult to interpret the latent space generated by MVA, as the links from the representation to the data through the decoder are highly nonlinear. One question we can ask is, given a focal brand, which brands are closest to it in the latent space? We show this analysis for four brands in Table 3.11.

Brands that are closeby in z_b space are predicted to have similar properties across the different modalities. In some cases, the results in Table 3.11 are very intuitive. For instance, McDonald's tends to be close to many mass market, affordable chain stores, with dense, simple logos, often operating in the food industry. Starbucks' closest neighbors share circular properties, as well as operating within the slightly upper scale food space. Nike's closest neighbor is Adidas, which is similar both in terms of aesthetics and function to Nike. Finally, Actavis, a pharmaceutical manufacturer, is close to other manufacturing and B2B firms, with again similar logos, especially in terms of font, color scheme, and mark complexity.

McDonalds and Supervalu To help build intuition about Table 3.11, let's consider the very first example: McDonalds, and its nearest neighbor Supervalu. As just described, McDonalds and Supervalu have many superficial similarities: they both have red, bold logos, and operate in the discount food space. These similarities are also reflected in the data. If we consider how people perceive these brands, vis-a-vis brand personality, there are huge similarities, as plotted in Figure 3.17. Moreover,

Focal Brand	Neighbors in z_b space			
McDonalds Fast food	**Supervalu** Retailing and grocery	**Old Navy** Apparel	**Dollar General** Discount retailer	**Kroger** Grocery
Starbucks Coffee	**Chipotle** Fast casual restaurant	**Whole Foods** Organic grocer	**L'Oreal** Personal care	**Minute Maid** Juice and beverage
Nike Footwear and apparel	**Adidas** Footwear and apparel	**Disney** Media and entertainment	**Polaris** Snowmobiles and ATVs	**Lego** Toys
Actavis Pharmaceutical manufacturing	**Praxair** Industrial gases	**Autoliv** Automotive safety supplier	**Clorox** Consumer products	**Optum Health** Health services

Table 3.11: The 4 closest brands to each focal brand in z_b space, including their logo, name, and a brief description.

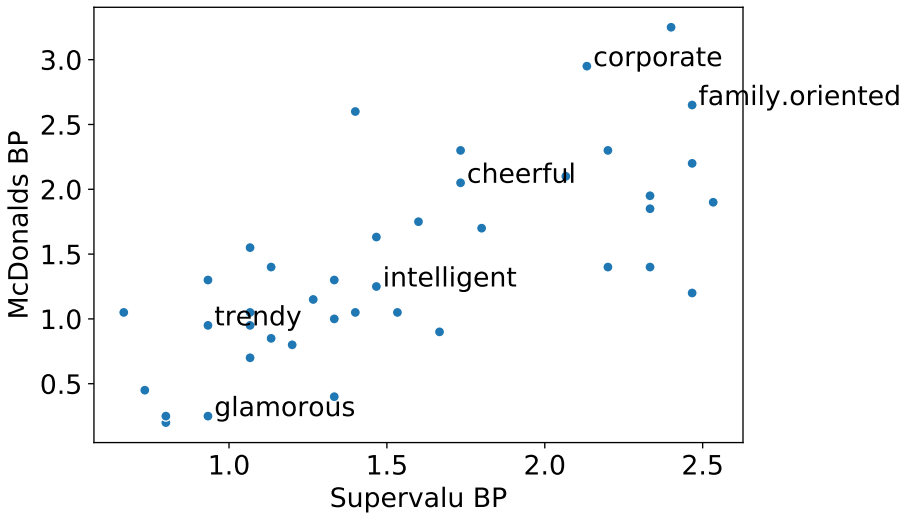

Figure 3.17: Brand personality ratings of McDonalds versus Supervalu.

the words that the two brands use to describe themselves are also similar: the correlation between McDonalds binary text vector and that of Supervalu is $r = 0.24$, nearly double the correlation of McDonalds with other firms on average ($r = 0.13$). These similarities across all modalities among these two brands is what the model is detecting, leading to their similar representations in z space. Similarity in z space then leads to similar predictions in all of the modalities.

Nike and Disney There are some less intuitive findings in Table 3.11 as well. Perhaps most interesting are Nike's neighbors besides Adidas: Disney, Polaris, and Lego. Let's consider the similarities between Nike and Disney. Aesthetically, their logos are, in fact similar, in terms of color and layout. Interestingly, their brand personalities are also aligned, as we show in Figure 3.18. What's striking about this plot is on how many dimensions both brands score near the top of the scale, including on dimensions like successful, imaginative, and family-oriented. There are also some differences, especially related to the ruggedness of Nike. Finally, the words they use to describe themselves are also similar: the correlation between

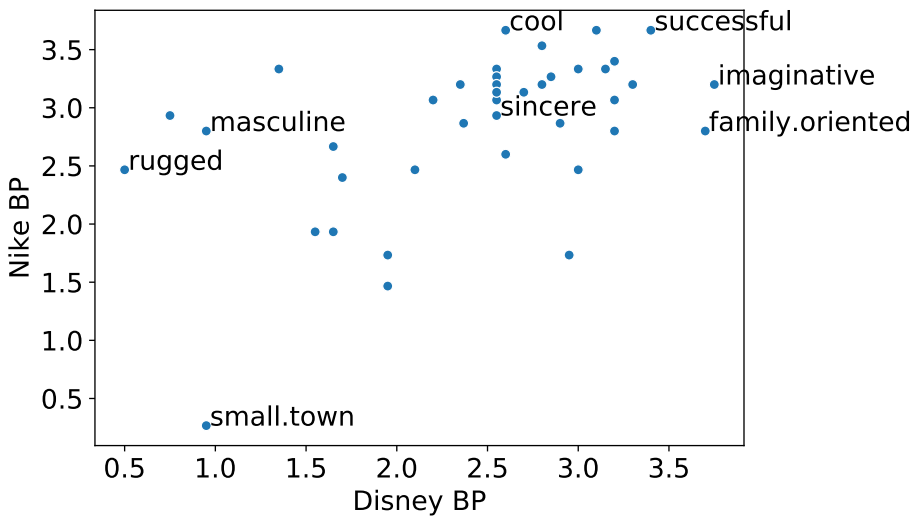

Figure 3.18: Brand personality ratings of Nike versus Disney.

Nike's binary word vector and that of Disney is $r = 0.2$, compared to $r = 0.12$ on average across all brands. Hence, while perhaps surprising at first glance, there are deep connections between the brand identities of Nike and Disney, which the model detects, and then subsequently predicts.

Interpolating Between Brands

Another way to attempt to understand the latent 10-dimensional space learned by our MVA is to use it to interpolate between brands. Intuitively, the MVA converts a large set of features with very different statistical properties into compact, continuous vector representations. Continuous movement in this latent space thus allows for continuous movement among brand identities, slowly shifting the predictions of the model. We can use such movement in the latent space to interpolate between brand identities.

Rank (Closest)	p = 0.9	p = 0.7	p = 0.5	p = 0.3	p = 0.1
1					
2					
3					

Table 3.12: Linear interpolation between McDonalds and Nike, showing three brands, in order, whose z_b are closest to $z = pz_{\text{McDonalds}} + (1-p)z_{\text{Nike}}$.

McDonalds and Nike Midpoint Analysis For instance, drawing on our previous analyses, we may ask the question: which brand identities emerge by interpolating between McDonalds and Nike? To answer this question, we consider new z values of the following form:

$$z = pz_{\text{McDonalds}} + (1-p)z_{\text{Nike}}.$$

We consider $p = 0.9, 0.7, 0.5, 0.3, 0.1$. We then find the actual z_b vectors that are closest to this interpolated value for each value of p. We show the results in Table 3.12. In general, we find a few transitions that happen between these identities: first, we see the apparel companies like Old Navy that were previously similar McDonalds emerge as the most similar to the interpolation. We also see the element of "value" fade away, as firms like Supervalu and Dollar General disappear, and firms like Gap appear. At the midpoint, we see Cadbury, a chocolate company, emerge as the midpoint. Finally, as we move toward Nike, we see Disney and Adidas again emerge, although Disney emerges sooner than Adidas.

It is interesting to consider why the model identifies Cadbury as the closest brand to the midpoint of McDonalds and Nike. First, it's worth noting that, while Cadbury is the closest in terms of z distance to the midpoint, it is not exactly at the midpoint. In fact, on several dimensions, the Cadbury z_b is actually quite far

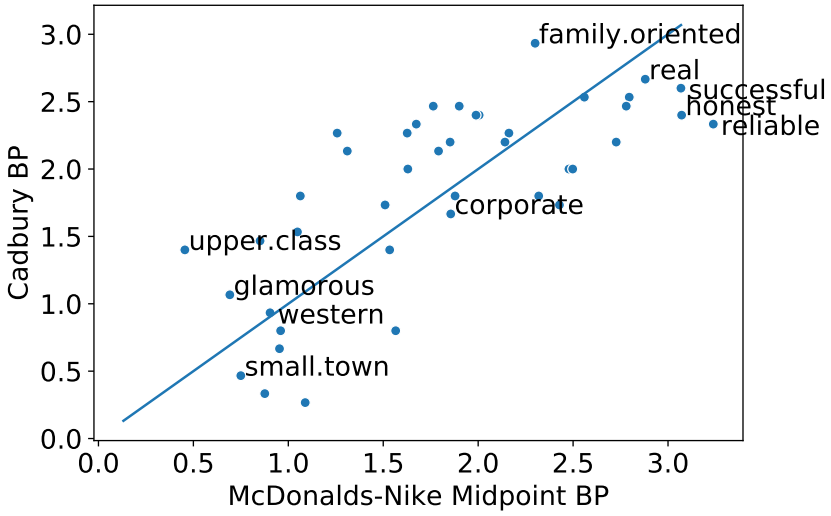

Figure 3.19: The actual brand personality of Cadbury is plotted against the predicted brand personality of the fictitious brand that lies at the midpoint between McDonalds and Nike in z space.

from the midpoint. Thus, in some sense, no brand exists at the exact midpoint between McDonalds and Nike.

There are, however, some clear similarities between this fictitious midpoint brand and Cadbury. In Figure 3.19, we plot the brand personality of Cadbury against what the model predicts for the fictitious midpoint brand. We see a close, but not exact correlation: the model predicts a midpoint brand that scores high on reliability, honesty, and success, while low on upper class, glamorous, and small town. Cadbury, on the other hand, does not score as low on things like upper class and glamorous, but agrees with much of the rest of the profile.

For visual features, the model predicts that the midpoint brand will have a very high degree of saturation in terms of colors (i.e. very dark, vivid colors), and a very high number of vertically oriented edges. It will have few corners, few diagonal edges, and very little whitespace. It predicts it will be both horizontally and vertically symmetric. More specifically in terms of color, it predicts it will have

fewer colors in general, and predicts shades of blue and other dark colors as the most likely. In terms of shape, it predicts both a square logo, and a square mark, although it is somewhat unlikely to have a mark at all. In terms of font, it predicts a bold, sans-serif font, with original character sizing. Visually, Cadbury appears to somewhat match this profile, insofar as it uses dark, vivid colors with minimal whitespace. It also features a curved design that is roughly symmetric both horizontally and vertically. However, many of the other predictions are off. Cadbury is ovular, not square, with a calligraphic font.

Finally, in terms of text, we can ask which words the model expects will occur at rates significantly above the base rate for the midpoint brand. In this case, it expects words like meaning, compete, step, footprint, citizen, force, healthier, dollar, happen, creation, and breakthrough. Intuitively, these words do appear to be a midpoint between McDonalds and Nike, emphasizing dollars, creating, health, competition, and footprint. We can also ask which words occur significantly below what the average rate. In this case, the model does not expect to find words like day, high, experience, can, create, employee, serve, best, focus, way, solution, deliver, company, and every. These are terms that tend to describe big corporations (solution, deliver, company, employee). While these textual profiles appear coherent, neither particularly describes Cadbury, whose website text emphasized concepts like manufacturing (manufacture, produce, deliver, scale), variety, price, quality, promise, and reputation.

Other Interesting Midpoints While by no means comprehensive, there are many other interesting findings like the above, which fall out of the model's ability to interpolate between brand identities. These include:

- Under Armour as the midpoint between Nike and Gucci: Under Armour is

positioned to some degree as an upscale fitness clothing brand. It thus makes sense that the midpoint between a fairly mainstream athletic brand, Nike, and a luxury fashion brand, Gucci, would be a brand like Under Armour.

- Booking and Priceline are at the midpoint between Google and Hyatt, emphasizing again this clean interpolation between brand identities and firm functionalities, with Booking and Priceline being search engines for hotels.
- eBay is at the midpoint between Amazon and Google, which is fascinating, given eBay's visual similarity to Google, but functional similarity to Amazon.
- Ralph Lauren is at the midpoint between Mercedes-Benz and Old Navy. Ralph Lauren is a more upscale and luxurious apparel brand, relative to Old Navy.

Implications for Design In the sense that the model allows for interpolation and "arithmetic" between brand identities, it mirrors the logo design process. Logo designers often start with a survey of an industry, competitors, and audience, and determine the key elements of design that convey meaning in each of these spaces. In coming up with a final design for a focal brand, the task is then one of interpolating: for instance, how do we think of the Starbucks of Chinese cuisine? The Uber of healthcare? How can we infuse a little bit of the brand identity of Gucci into the fast food industry? By being able to formulate such questions mathematically, as vector operations in a latent space, we make this process of interpolation data-driven.

3.7.3 Generating Brand Identities

Our MVA is what is known as a deep generative model in machine learning. This term arises because the model can be used to generate data that mirrors the input

data. Generation under the model simply involves sampling new zs from the prior, $z_k \sim \mathcal{N}(0, 1)$, then passing these new z vectors through the decoder networks.

Thus, one way in which the framework can be used is the random generation of brand identities, including visual and textual components. Such generation gives us several insights: first, it allows us to further explore the structure the model has learned, by seeing what brand identities it generates. Second, it can provide a mechanism for idea generation, as the simulated brands may be structured fusions of the input data. Finally, it gives us a way of validating the model: by randomly sampling brand identities from the model, then comparing them to brand identities generated from other methods, we can assess whether the model is capturing patterns that are relevant to consumer perceptions, and begin exploring what makes designs optimal, rather than typical. This latter point is beyond the scope of the current work, but is a direction for on-going research. For now, we illustrate the ability of the model to generate new brand identities.

Case Study: Randomly Generating a Cold, Modern Corporation

The randomly sampled z vector we we will examine is:

$$z = (0.61,\ 1.24,\ 0.96,\ 1.55,\ -0.26,\ -0.17,\ 2.96,\ -1.09,\ -0.48,\ -0.68).$$

As we can see from the model's predictions, this z corresonds to a brand identity which we label, "cold, modern corporation." The predicted brand personality profile corresponding to this z is displayed in Table 3.13. From this, we high scores on up-to-date, imaginative, technical, and corporate, and low scores on wholesome, sentimental, tough, and family-oriented. Together, this paints a picture of a technical, modern corporation.

Trait	Pred	Diff	Trait	Pred	Diff
up-to-date	2.81	0.37	leader	2.46	-0.12
imaginative	2.16	0.36	glamorous	0.89	-0.14
technical	2.35	0.33	smooth	1.49	-0.16
corporate	2.80	0.24	cheerful	1.49	-0.18
intelligent	2.72	0.24	reliable	2.44	-0.20
exciting	1.77	0.17	western	1.06	-0.22
spirited	2.07	0.17	friendly	1.76	-0.25
confident	2.87	0.16	sincere	1.96	-0.29
independent	2.38	0.15	charming	1.07	-0.32
daring	1.73	0.10	feminine	0.46	-0.37
secure	2.61	0.08	down-to-earth	1.35	-0.39
young	1.17	0.08	good looking	1.25	-0.40
contemporary	1.91	0.06	real	2.10	-0.41
trendy	1.62	0.05	small town	0.21	-0.46
upper class	1.49	0.01	family-oriented	1.26	-0.48
unique	1.78	-0.02	outdoorsy	0.61	-0.50
honest	2.32	-0.06	tough	0.94	-0.54
successful	2.89	-0.09	rugged	0.49	-0.59
original	1.89	-0.09	masculine	0.95	-0.60
hard working	2.47	-0.10	sentimental	0.42	-0.71
cool	1.57	-0.11	wholesome	0.94	-0.76

Table 3.13: The predicted brand personality profile for our first randomly generated brand, the "cold, modern corporation." We show both the predicted values from the model ("Pred"), and how those values differ from the data average ("Diff").

The words that the model predicts are most likely to appear on the brand's website are displayed in a word cloud in Figure 3.20. In addition to the top predicted words, we also show the words that are relatively likely and relatively unlikely for the simulated brand. In general, there are certain words that many firms use, including product, business, customer, world, provide, and service, which may not be as relevant to understanding the focal brand. We see that these two word clouds support the identity conveyed by brand personality: among the relatively likely words, we find technical words like data, app, problem, and implement. In the relatively unlikely words, we find things like provide, family, culture, and life.

Finally, we can see the visual features we expect to find in this firm's logo by

Most Likely

Relatively Likely

Relatively Unlikely

Figure 3.20: At top, a random sample of the words that the model predicts will occur with greater than 50% probability, drawn proportional to their probability. At bottom left, the words that the model predicts will occur significantly more than they occur on average. At bottom right, the words that the model predicts will occur significantly less than they occur on average.

Color		Font		Layout	
Feature	Prob	Feature	Prob	Feature	Prob
Has: blue dark	0.671	Font: wide	0.823	Has mark	1.000
Has: blue medium	1.000	Font: bold	0.956	Mark pos: bottom	0.967
Has: yellow	0.992	Font: no italics	0.981	Mark pos: top	0.556
Accent: blue medium	0.995	Class: geometric square	0.744		
Accent: yellow	0.998	Class: clarendon	0.525		

Table 3.14: Binary logo features that the model predicted would occur with greater than 50% probability for the generated brand, together with the predicted probabilities.

Feature	Value	Feature	Value
# Characters	-0.46	Aspect Ratio	-1.31
# Colors	-0.30	Entropy	0.33
# Corners	-0.33	Perimetric Complexity	-0.71
# Marks	-0.27	Horizontal Symmetry	0.50
# Regions	-0.54	Vertical Symmetry	-0.90
% Whitespace	-0.92	Mean Lightness	-0.75
Vertical Edges	0.39	Mean Saturation	0.53
Down Diag Edges	-0.16	SD Lightness	0.44
Horizontal Edges	-0.28	SD Saturation	0.55
Up Diag Edges	-0.30		

Table 3.15: Real-valued logo features that the model predicted for the generated brand. These values are standardized values (z-scores), and hence can be interpreted as standard deviations different from the average value of the feature.

examining Tables 3.14, 3.15, and 3.16. An interpretation of this logo by the author is presented in Figure 3.21.[12] It is harder to objectively interpret these visual elements, but we claim that this logo template appears to share similar elements to other logos in, for instance, the technology space.

Simulating More Identities

Generating identities from the model is straightforward. In this section, we present several additional simulations, albeit in less detail that the cold, modern

[12]The author is not a designer, as may be obvious from the interpretation.

Feature	First (Prob)	Second (Prob)	Third (Prob)
Dominant Color	Med. Blue (0.999)	Dark Blue (0.001)	Yellow (0.000)
Hull Class	Circle (0.652)	Triangle (0.293)	Med. Rect./Oval (0.028)
Mark Class	Wispy Horiz. (0.847)	Circular (0.113)	Square (0.023)
Font Serifs	Sans-Serif (0.671)	Serif (0.329)	No Chars (0.000)

Table 3.16: Predicted categorical logo features for the generated brand. For each feature, we list the top three most likely outcomes under the model, together with their probabilities. (Throughout, the abbreviation "Med." stands for "Medium."

Figure 3.21: A rendering by the author of a logo matching the features described in Tables 3.14, 3.15, and 3.16.

corporation above. Each of these was generated simply by evaluating each of the decoder network at a vector of 10 standard normal draws.

Sophisticated Media The following corresponds to a brand identity with $z = (-1.60, 0.45, -0.71, -1.35, -1.29, 1.50, -1.36, 0.01, 1.23, -1.33)$:

- Relatively likely words: 'book', 'physic', 'televis', 'word', 'step', 'decemb', 'sophist', 'someth', 'pleas', 'readi'
- Relatively unlikely words: 'communiti', 'can', 'custom', 'compani', 'global', 'solut', 'servic', 'innov', 'work', 'provid'
- Top three relative brand personality traits: glamorous, trendy, exciting
- Bottom three relative brand personality traits: masculine, hard working, wholesome

- Some likely visual features: black dominant color, yellow and light green accent colors, light font, no italics, geometric font class, has a mark

From these traits, we label this a sophisticated media firm.

Family Friendly Food The following corresponds to a brand identity with $z = (-1.12, 0.22, 0.04, -1.22, 1.17, 0.56, 0.28, 0.91, 1.11, 0.83)$:

- Relatively likely words: 'www', 'televis', 'central', 'happen', 'mutual', 'dollar', 'ingredi', 'ultim', 'hand', 'kind'
- Relatively unlikely words: 'employe', 'technolog', 'solut', 'global', 'new', 'custom', 'work', 'innov', 'servic', 'provid'
- Top three relative brand personality traits: cheerful, friendly, family-oriented
- Bottom three relative brand personality traits: rugged, tough, masculine
- Some likely visual features: brown dominant color, red and yellow accent colors, bold font, geometric font class, has a mark

From these traits, we label this a family-friendly food firm.

3.7.4 Crossmodal Inferences

Finally, from a decision support perspective, the most critical component of our model is the ability to move across modalities. That is, to predict, for instance, a logo, from a textual brief. This allows us to inform the design process in a data-driven fashion, by automatically translating text and survey data into visual templates. The ability to go from a logo to text and personality is also important,

insofar as it allows for both the evaluation of potential identities, and for "letting the logos speak," to gain a better understanding of common design patterns. In this section, we illustrate the two former channels: going from brand personality to text and logo, and going from text to brand personality and logo.

Before delving into those illustrations, we describe how the process works, mathematically. In all cases, crossmodal predictions work through the modality specific inference networks, combined with the full decoder network. Specifically, given data on domain d for a new firm, denoted x^d_{new}, we can learn the approximate posterior of that brand's representation, z_{new}, via the modality d inference network:

$$z_{\text{new}} \sim \mathcal{N}(\xi_{\text{new},1}, \xi_{\text{new},2}),\ \xi_{\text{new}} = \text{INet}_d(x_{\text{new}_\text{d}}; \phi_d).$$

We can then make predictions for any of the domains by passing the expectation for z_{new}, $E(z_{\text{new}}) = \xi_{\text{new},1}$ through the decoder network for any of the domains of interest, d^*:

$$p(x^{d^*}_{\text{new}} \mid z_{\text{new}}) = p(x^{d^*}_{\text{new}} \mid \mu^{d^*}_{bj} = \text{DNet}_{d^*}(z_{\text{new}}; \theta_{d^*})).$$

This reveals the practicality of this multiview inference network approach: evaluating a conditional posterior predictive is equivalent to evaluating two functions: the inference network of the given domain d to infer the posterior of the latent parameter z, and the decoder network of the domain of interest d^*, conditional on the inferred z.

Brand Personality to Textual and Visual Identity

Given a brand personality profile, our goal is to use the MVA framework to understand what words might describe a firm with that personality, and what features are likely in that firm's logo. As a case study, we will focus on a firm that is

a rugged, masculine, reliable, and hard working firm, with brand personality profile (relative to the mean) displayed in Figure 3.22. The brand personality inference network computes an expected z for this brand as:

$$z = (-0.06, 1.75, -0.52, 0.41, 0.52, -0.47, 0.93, -0.26, -0.93, -0.30).$$

Plugging this z into the text decoder network, we find the most likely words are those shown visually in the word cloud in Figure 3.23, and in Table 3.17. Visually, the model expects to find again a blue logo, similar to the randomly generated firm in the previous section. The accent colors it expects now are again yellow, but also light blue. The font it expects is distinct from the random profile: it expects that this firm will use bold condensed letters, as opposed to wide. In terms of convex hull, it gives the highest probability to a circular or wide ovular/rectangular logo. Finally, similar to the random logo, it expects this firm will have a dark logo with low whitespace, and with a lower than average aspect ratio, indicating that it is less wide and more tall than average. We again provide a non-professional rendering of a logo meeting many of these criteria in Figure

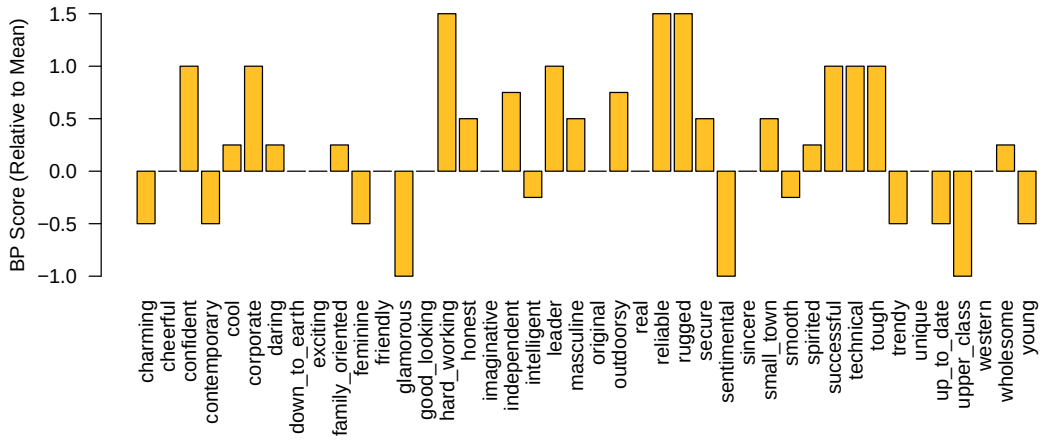

Figure 3.22: Brand personality of our focal firm for doing crossmodal inferences. Personality values are shown relative to the mean (i.e. differenced from the mean personality value across all firms).

Figure 3.23: A word cloud reflecting the most likely words generated from the crossmodal inference procedure for the focal brand personality profile, corresponding to words that would likely be on the website of a firm with that brand personality.

	Top 20 Words
Most Likely	promis, regul, unit, whole, men, shop, accomplish, effici, specialti, women, account, environment, strong, solut, mobil, visit, exceed, divis, heritag, abil
Relatively Likely	regul, ceo, meaning, compet, scientif, whole, treatment, footprint, sector, trend, dollar, forc, implement, latest, faster, healthier, everywher, clinic, sophist, compon
Relatively Unlikely	improv, compani, time, experi, state, high, around, deliv, also, day, offer, countri, best, can, everi, creat, provid, us, new, work

Table 3.17: Likely and unlikely words generated from the crossmodal inference procedure for the focal brand personality profile, corresponding to words that would (not) be on the website of a firm with that brand personality.

Figure 3.24: Rendering of a logo containing many of the traits the model predicts given the focal brand personality.

Text to Logo and Brand Personality

Our final illustration of crossmodal inferences illustrates the direction that most approximates the design process: from a textual description to a logo and a prediction of brand personality perceptions. For this section, we will focus on a firm that was not included in our original dataset: Shake Shack. Shake Shack is a modern fast casual restaurant, serving burgers, hot dogs, milkshakes, and french fries, based out of New York City. We processed this text as we did the brands in our original sample, and present a summary of the text from their website in Figure 3.25. We then used the text inference network to infer Shake Shack's latent z_b. This was then passed to our logo and personality inference networks, to predict the feature's of Shake Shack's logo, and the way consumers will perceive their brand personality.

In Figure 3.26, we present the brand personality predictions, which we assess to be relatively accurate: Shake Shack is a fairly trendy, contemporary take on fast food. It is generally perceived as (relatively) glamorous and exciting, especially in its association with New York City, and cheerful in both what it does, and how it portrays itself. In Tables 3.18, 3.19, and 3.20, we give the logo predictions, which are somewhat less accurate. Interestingly, the accent color of (light) green is accurately predicted, as is the square font, the high perimetric complexity, the vertical symmetry, and the higher variation in lightness. But many of the other predictions, including the dominant color of brown, the bold font, and the left placement of the mark are off.

We may then ask, why do the model's predictions differ from reality in the case of Shake Shack? One interpretation is that Shake Shack has intentionally deviated from the mold, to draw on certain poignant associations. For instance, an

interesting element of the Shake Shack logo is its resemblance to an old neon sign, emblematic of an old school hot dog stand. While a standard fast food or fast casual restaurant may indeed feature bold font, Shake Shack differs from the model to emphasize its heritage. Moreover, the emphasis on blacks, instead of browns or blues, is characteristic of sophistication, charm, and luxury. Originating in New York, and marketing itself as a more upscale experience, relative to the standard burger and fries chains, perhaps this relative empahsis is thus strategic in its appeal to the New York demographic, and its key point of differentiation from the competition.

Figure 3.25: A word cloud representing the text from Shake Shack's website, processed in the same way as our original textual data.

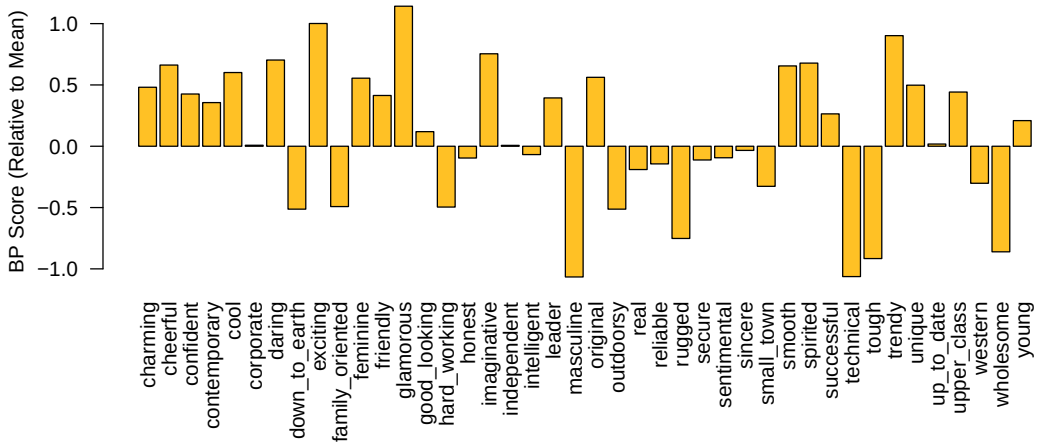

Figure 3.26: Brand personality predictions for Shake Shack, relative to the mean brand personalities in our sample, based on a crossmodal prediction from Shake Shack's text.

Color		Font		Layout	
Feature	Prob	Feature	Prob	Feature	Prob
Has: Dark Blue	0.729	Weight: Bold	0.746	Mark pos: Left	0.509
Has: Med Blue	0.826	Weight: Original	0.654	Has Mark	1.000
Has: Light Green	0.861	No Italics	1.000		
Accent: Dark Blue	0.820	Class: Geometric Square	1.000		
Accent: Light Blue	0.923				
Accent: Light Green	0.959				

Table 3.18: Binary logo features that the model predicted would occur with greater than 50% probability for Shake Shack, together with the predicted probabilities.

Feature	Value	Feature	Value
# Characters	0.22	Aspect Ratio	0.37
# Colors	0.59	Entropy	0.52
# Corners	0.05	Perimetric Complexity	0.60
# Marks	-0.26	Horizontal Symmetry	-0.47
# Regions	-0.01	Vertical Symmetry	0.87
% White	-0.37	Mean Lightness	-0.33
Vertical Edges	1.26	Mean Saturation	-0.28
Down Diag Edges	0.50	SD Lightness	0.15
Horizontal Edges	-0.75	SD Saturation	-0.56
Up Diag Edges	-1.14		

Table 3.19: Real-valued logo features that the model predicted for the Shake Shack. These values are standardized values (z-scores), and hence can be interpreted as standard deviations different from the average value of the feature.

Feature	First (Prob)	Second (Prob)	Third (Prob)
Dominant Color	Brown (0.847)	Med. Blue (0.141)	Dark Blue (0.006)
Hull Class	Med. Rect./Oval (0.665)	Thin Rect./Oval (0.333)	Triangle (0.001)
Mark Class	Vertical Narrow (0.359)	Square (0.330)	Bulky Hollow Geom. (0.306)
Font Serifs	Sans-Serif (0.970)	Serif (0.030)	No Characters (0.000)

Table 3.20: Predicted categorical logo features for Shake Shack. For each feature, we list the top three most likely outcomes under the model, together with their probabilities. (Throughout, the abbreviation "Med." stands for "Medium."

3.8 Conclusions and Ongoing Work

In this work, we have explored logo design and brand identity from a data-driven perspective. Our primary contributions are an approach to working with logos as data in a way that is both automatic and human interpretable, predictive results which can help identify specific features of interest and understand patterns in design, and finally a multiview learning model that mimics the design process, and in which we introduce a new approach for using variational autoencoders for multiview learning. Our feature extraction algorithm makes the process of understanding logo design both objective, in the sense that it is done automatically through image processing, and useful for designers, in the sense that the features are interpretable. The multiview learning model provides a way of moving across distinct modalities, including text, logos, and brand personality, to aid in the design process. We introduced a novel inference algorithm that mimics the way in which the model is meant to be used. Finally, in applying the model to our data, we learned a latent space that is meaningful, and in which vector operations, like the interpolation between two brands, yields interesting insights to brand identity.

This project is still on-going. Our continuing research agenda includes four key items:

1. First, the set of features included in the multiview learning model is still incomplete. We are not yet taking advantage of the industry tags, or the full set of visual features which describe the marks themselves. Marks are an important aspect of logo design, and incorporating this information will likely yield a richer picture of the design process.

2. Second, more study needs to be done on the network architectures. In

particular, a point of interest is understanding the weight that the different domains are given in the learned representation, and if this is affected by the network structures used.

3. Third, we need more stringent measures of validation. While the inference procedure naturally involves prediction and dimensionality reduction, thus seemingly alleviating potential overfitting problems, a better set of holdout metrics is needed. This is particularly important in light of some of the strong probabilities that the model asserts for certain features (e.g. words occurring with probability 1).

4. Finally, related to the third point, and perhaps most importantly, we want to validate these model results on true consumers. For this, we seek to run lab studies in which consumers are presented with logos generated from this framework, versus logos that are either random, or assembled in some other way. Then, consumers will evaluate the logos based on some measures of typicality (e.g. how well the logo matches a textual description). This kind of experiment can then be used to validate the model predictions in the most meaningful way: consumer perceptions.

Finally, there are several important limitations of this study. Foremost, our model is a model of typicality, not optimality, as alluded to particularly with the example of Shake Shack. We are able to capture what a typical firm does, not what is the best logo for a firm to do, given certain objectives other than typicality. Additionally, our model does not make strong claims about the causality of design: that is, why are existing logos designed the way they are? Answering this question is difficult, and likely involves both temporal factors (e.g. mimicry of a successful brand) and functional factors (e.g. red is easy to see on a sign from far away, or red stimulates the appetite). We leave both of these issues as topics for future study.

3.9 Appendix: Logo Feature Details

Category	Feature	Level	Description	Value	Literature
Color	Color	Both	Whether a given color is present	Binary	Valdez and Mehrabian (1994); Klink (2003); Deng et al. (2010); Semin and Palma (2014); Kareklas et al. (2014)
	Dominant Color	Both	The color with the highest number of pixels	Categorical	
	Accent Color	Both	All colors that are not the dominant color	Binary	
	% Whitespace	Both	How much of the logo (mark)'s convex hull is background (whitespace)?	Real	
	Mean Saturation	Both	The mean value of the saturation channel across pixels in HSV colorspace	Real	
	SD Saturation	Both	The standard deviation of the saturation channel	Real	
	Mean Lightness	Both	The mean value of the value channel in HSV colorspace	Real	
	SD Lightness	Both	The standard deviation of the value channel	Real	
Format and Shape	Has Mark	Global	Is there a mark?	Binary	Navon (1977); Klink (2003); Orth and Malkewitz (2008); Walsh et al. (2010) Spence (2012)
	Size	Mark	How much of the logo does the mark take up	Real	
	Number of Marks	Global	How many marks there are	Count	
	Convex hull	Global	The smallest convex polygon that fully contains the logo, classified into types	Categorical	

	Standardized shape	Mark	The mark is standardized into a 25×25 pixel shape, then clustered pixelwise, weighted by size, which captures similarity in both shape and size of the mark	Categorical	
	Aspect Ratio	Both	The ratio of the height and width	Real	
	# Corners	Both	The number of corners found by the Harris corner detector	Count	
Font	# Characters	Global	Number of logo segments classified as characters	Count	Doyle and Bottomley (2004); Henderson et al. (2004)
	Serif	Global	Classification of characters into serif, sans-serif, or calligraphic fonts	See footnote[13]	
	Family	Global	Vox-ATypI font families		
	Italics	Global	Upright versus italic characters		
	Weight	Global	Original, bold, or light characters		
	Width	Global	Original, condensed, or wide characters		
Complexity	# Colors	Both	How many distinct colors are there?	Count	Henderson and Cote (1998);
	# Segments	Both	How many distinct regions are there?	Count	Janiszewski et al. (2001);

[13]The basic type of all font variables is a count: for every identified character, we match it to one element in our font dictionary, which then determines all of the font properties. Thus, the basic feature is a count of how many times each font feature appears (e.g. 5 bold letters, 4 geometric fonts). However, noting that this matching is just a noisy approximation of the font features, we also form features from these counts. For example, we may model the dominant font family, or sans versus serif, as a categorical variable, where the outcome is the family or type with the highest count.

	Perimetric complexity	Both	A measure of shape complexity, given by the ratio of the number of edge pixels to interior pixels, where the edge pixels are computed via canny edge detection	Real	van der Lans et al. (2009); Pieters et al. (2010)
	Greyscale entropy	Both	The local average variance of greyscale pixel intensity	Real	
Symmetry	Horizontal Symmetry	Both	The correlation in pixel values when the image (mark or logo) is split in half horizontally (i.e. left and right halves)	Real	Henderson and Cote (1998); van der Lans et al. (2009)
	Vertical Symmetry	Both	The correlation in pixel values when the image (mark or logo) is split in half vertically (i.e. top and bottom halves)	Real	
Repetition	Size Repitition	Both	The standard deviation of the sizes of the subcomponents of the image (mark or logo)	Real	Henderson and Cote (1998); van der Lans et al. (2009)
	Complexity Repitition	Both	The standard deviation of the perimetric complexity of the subcomponents of the image (mark or logo)	Real	

Orientation	Position	Both	The position of the mark relative to the text. We compute both hard and soft versions of this metric: for example, hard left means the mark is entirely to the left of the text, whereas soft left means that the center of the mark is to the left of the center of the text.	Binary	Chae and Hoegg (2013); Cian et al. (2014);
	Edge Gradients	Both	The percentage of non-zero edge gradients classified as horizontal, vertical, up-diagonal, or down-diagonal, computed by traversing the binarized logo in both left-right and top-down directions and computing numerical gradients.	Real	Deng and Kahn (2016); Schlosser et al. (2016)

Bibliography

Aaker, J. L. (1997). Dimensions of Brand Personality. *Journal of Marketing Research*, 34(3):347.

Ansari, A. and Iyengar, R. (2006). Semiparametric Thurstonian models for recurrent choices: a Bayesian analysis. *Psychometrika*, pages 631–657.

Ansari, A. and Mela, C. F. (2003). E-Customization. *Journal of Marketing Research*, 40(2):131–145.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877.

Bockenholt, U. (2006). Thurstonian-based analyses: Past, present, and future utilities. *Psychometrika*, 71(4):615–629.

Bronnenberg, B. J., Kruger, M. W., and Mela, C. F. (2008). The IRI Marketing Data Set. *Marketing Science*, 27(4):745–748.

Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M. A., Li, P., and Riddell, A. (2016). Stan: A Probabilistic Programming Language. *Journal of statistical software*.

Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M. A., Li, P., and Riddell, A. (2017). Stan: A Probabilistic Programming Language. *Journal of Statistical Software*, 76(1).

Cha, W. M., Chintagunta, P. K., and Dhar, S. K. (2015). Food Purchases During the Great Recession. *SSRN ID #2548758*.

Chae, B. G. and Hoegg, J. (2013). The Future Looks "Right": Effects of the Horizontal Location of Advertising Images on Product Attitude. *Journal of Consumer Research*, 40(August):223–238.

Childers, T. L. and Jass, J. (2002). All Dressed Up With Something to Say: Effects of Typeface Semantic Associations on Brand Perceptions and Consumer Memory. *Journal of Consumer Psychology*, 12(2):93–106.

Chintagunta, P., Dubé, J.-P., and Goh, K. Y. (2005). Beyond the Endogeneity Bias: The Effect of Unmeasured Brand Characteristics on Household-Level Brand Choice Models. *Management Science*, 51(5):832–849.

Cian, L., Krishna, A., and Elder, R. S. (2014). This logo moves me: Dynamic imagery from static images. *Journal of Marketing Research*, 51(2):84–197.

Damianou, A. C. and Lawrence, N. D. (2013). Deep Gaussian Processes. *International Conference on Artificial Intelligence and Statistics*, 31:207–215.

Deng, X., Hui, S. K., and Hutchinson, J. W. (2010). Consumer preferences for color combinations: An empirical analysis of similarity-based color relationships. *Journal of Consumer Psychology*, 20(4):476–484.

Deng, X. and Kahn, B. E. (2016). Is Your Product on the Right Side? The "Location Effect" on Perceived Product Heaviness and Package Evaluation. *Journal of Marketing Research*, (Forthcoming).

DeSarbo, W. S., Ansari, A., Chintagunta, P. K., Himmelberg, C., Jedidi, K., Johnson, R., Kamakura, W., Lenk, P., Srinivasan, K., and Wedel, M. (1997). Representing Heterogeneity in Consumer Response Models. *Marketing Letters*, 8(3):335–348.

DeSarbo, W. S., Fong, D. K. H., Llechty, J., and Coupland, J. C. (2005). Evolutionary preference/utility functions: A dynamic perspective. *Psychometrika*, 70(1):179–202.

Dew, R. and Ansari, A. (2016). Bayesian Nonparametric Customer Base Analysis with Model-based Visualizations. *Marketing Science*, (Forthcoming).

Dew, R. and Ansari, A. (2018). Bayesian Nonparametric Customer Base Analysis with Model-based Visualizations. *Marketing Science*, 37(2).

Dieng, A. B., Kim, Y., Rush, A. M., and Blei, D. M. (2018). Avoiding Latent Variable Collapse With Generative Skip Models.

Doyle, J. R. and Bottomley, P. A. (2004). Font appropriateness and brand choice. *Journal of Business Research*, 57(8):873–880.

Doyle, J. R. and Bottomley, P. A. (2006). Dressed for the Occasion: Font-Product Congruity in the Perception of Logotype. *Journal of Consumer Psychology*, 16(2):112–123.

Durbin, J. and Koopman, S. J. S. (2012). *Time series analysis by state space methods*.

Duvenaud, D., Lloyd, J., Grosse, R., Tenenbaum, J., and Ghahramani, Z. (2013). Structure discovery in nonparametric regression through compositional kernel search. *Proceedings of the International Conference on Machine Learning (ICML)*, 30:1166–1174.

Duvvuri, S. D., Ansari, A., and Gupta, S. (2007). Consumers' Price Sensitivities Across Complementary Categories. *Management Science*, 53(12):1933–1945.

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pages 226–231.

Fader, P., Hardie, B., and Lee, K. L. (2005). Counting Your Customers the Easy Way: An Alternative to the Pareto/NBD Model. *Marketing Science*, 24(2):275–284.

Fader, P., Hardie, B., and Shang, J. (2010). Customer-Base Analysis in a Discrete-Time Noncontractual Setting. *Marketing Science*, 29(6):1086–1108.

Feng, F., Wang, X., and Li, R. (2014). Cross-modal Retrieval with Correspondence Autoencoder. *Proceedings of the ACM International Conference on Multimedia - MM '14*, pages 7–16.

Flaxman, S., Gelman, A., Neill, D., Smola, A., and Vehtari, A. (2016). Fast hierarchical Gaussian processes.

Fuglstad, G.-A., Simpson, D., Lindgren, F., and Rue, H. (2018). Constructing priors that penalize the complexity of gaussian random fields. *Journal of the American Statistical Association*, pages 1–8.

Gelman, A. and Rubin, D. B. (1992). Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science*, 7(4):457–511.

Gilboa, E., Saatci, Y., and Cunningham, J. P. (2015). Scaling multidimensional inference for structured gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):424–436.

Gordon, B., Goldfarb, A., and Li, Y. (2013). Does Price Elasticity Vary with Economic Growth? A Cross-category Analysis. *Journal of Marketing Research*, 50(February):4–23.

Guadagni, P. M. and Little, J. D. C. (1983). A Logit Model of Brand Choice Calibrated on Scanner Data. *Marketing Science*, 2(3):203–238.

Guhl, D., Baumgartner, B., Kneib, T., and Steiner, W. J. (2018). Estimating time-varying parameters in brand choice models: A semiparametric approach. *International Journal of Research in Marketing*, 35(3):394–414.

Gupta, S. (1991). Stochastic models of interpurchase time with time-dependent covariates. *Journal of Marketing Research*, 28:1–15.

Hagtvedt, H. (2011). The Impact of Incomplete Typeface Logos on Perceptions of the Firm. *Journal of Marketing*, 75(4):86–93.

Hanssens, D. M., Parsons, L. J., and Schultz, R. L. (2001). *Market Response Models: Econometric and Time Series Analysis*. Kluwer Academic Publishers, 2nd edition.

Henderson, P. W. and Cote, J. A. (1998). Guidelines for Selecting or Modifying Logos. *Journal of Marketing*, 62:14–30.

Henderson, P. W., Cote, J. A., Leong, S. M., and Schmitt, B. (2003). Building strong brands in Asia: Selecting the visual components of image to maximize brand strength. *International Journal of Research in Marketing*, 20:297–313.

Henderson, P. W., Giese, J. L., and Cote, J. A. (2004). Impression Management Using Typeface Design. *Journal of Marketing*, 68(4):60–72.

Hoffman, M. and Gelman, A. (2014a). The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15:1351–1381.

Hoffman, M. and Gelman, A. (2014b). The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15:30.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.

Janiszewski, C. and Meyvis, T. (2001). Effects of Brand Logo Complexity, Repetition, and Spacing on Processing Fluency and Judgment. *Journal of Consumer Research*, 28(1):18–32.

Jerath, K., Fader, P. S., and Hardie, B. G. (2011). New Perspectives on Customer "Death" Using a Generalization of the Pareto/NBD Model. *Marketing Science*, 30(5):866–880.

Jiang, Y., Gorn, G. J., Galli, M., and Chattopadhyay, A. (2015). Does Your Company Have The Right Logo? How and Why Circular and Angular Logo Shapes Influence Brand Attribute Judgments. *Journal of Consumer Research*, 42:ucv049.

Kalyanam, K. and Shively, T. S. (1998). Estimating Irregular Pricing Effects: A Stochastic Spline Regression Approach Estimating Irregular Pricing Effects: A Stochastic Spline Regression Approach. *Journal of Marketing Research*, 35(1):16–29.

Kareklas, I., Brunel, F. F., and Coulter, R. A. (2014). Judgment is not color blind: The impact of automatic color preference on product and advertising preferences. *Journal of Consumer Psychology*, 24(1):87–95.

Kaufman, C. G. and Shaby, B. A. (2013). The role of the range parameter for estimation and prediction in geostatistics. *Biometrika*, 100(2):473–484.

Kim, J. G., Menzefricke, U., and Feinberg, F. M. (2004). Assessing Heterogeneity in Discrete Choice Models Using a Dirichlet Process Prior. *Review of Marketing Science*, 2(1):1–39.

Kim, J. G., Menzefricke, U., and Feinberg, F. M. (2005). Modeling Parametric Evolution in a Random Utility Framework. *Journal of Business & Economic Statistics*, 23(3):282–294.

Kim, J. G., Menzefricke, U., and Feinberg, F. M. (2007). Capturing Flexible Heterogeneous Utility Curves: A Bayesian Spline Approach. *Management Science*, 53(2):340–354.

Kingma, D. P. and Welling, M. (2013). Auto-Encoding Variational Bayes. (Ml):1–14.

Klink, R. R. (2003). Creating Meaningful Brands: The Relationship Between Brand Name and Brand Mark. *Marketing Letters*, 14(3):143–157.

Lachaab, M., Ansari, A., Jedidi, K., and Trabelsi, A. (2006). Modeling preference evolution in discrete choice models: A Bayesian state-space approach. *Quantitative Marketing and Economics*, 4(1):57–81.

Li, Y. and Ansari, A. (2014). A Bayesian Semiparametric Approach for Endogeneity and Heterogeneity in Choice Models. *Management Science*, 60(5):1161–1179.

Li, Y., Yang, M., and Zhang, Z. (2016). Multi-View Representation Learning: A Survey from Shallow Methods to Deep Methods. 14(8):1–20.

Liechty, J. C., Fong, D. K. H., and DeSarbo, W. S. (2005). Dynamic Models Incorporating Individual Heterogeneity: Utility Evolution in Conjoint Analysis. *Marketing Science*, 24(March 2015):285–293.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee.

McLaren, K. (1976). The Development of the CIE 1976 (L* a* b*) Uniform Colour Space and Colour-difference Formula. *Journal of the Society of Dyers and Colourists*, 92(9):338–341.

Mela, C. F., Gupta, S., and Lehmann, D. R. (1997). The Long-Term Impact of Promotion and Advertising on Consumer Brand Choice. *Journal Of Marketing Research*, 34(2):248–261.

Meyers-Levy, J. and Peracchio, L. A. (1992). Getting an angle in advertising: The effect of camera angle. *Journal of Marketing Research*, 29(4):454–461.

Navon, D. (1977). Forest before trees: The precedence of global features in visual perception. *Cognitive Psychology*, 9(3):353–383.

Neal, R. M. (1998). Regression and Classification Using Gaussian Process Priors. *Bayesian Statistics*, 6:475–501.

Neal, R. M. (2011). MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, pages 113–162.

Neelamegham, R. and Chintagunta, P. K. (2004). Modeling and Forecasting the Sales of Technology Products. *Quantitative Marketing and Economics*, 2(3):195–232.

Neslin, S. A., Gupta, S., Kamakura, W., Lu, J., and Mason, C. H. (2006). Defection Detection: Measuring and Understanding the Predictive Accuracy of Customer Churn Models. *Journal of Marketing Research*, 43(2):204–211.

Ngiam, J., Khosla, A., and Kim, M. (2011). Multimodal Deep Learning. In *Proceedings of the 28th International Conference on Machine Learning*.

Orth, U. R. and Malkewitz, K. (2008). Holistic Package Design and Consumer Brand Impressions. *Journal of Marketing*, 72(3):64–81.

Patrick, V. M. and Hagtvedt, H. (2011). Aesthetic Incongruity Resolution. *Journal of Marketing Research (JMR)*, 48(2):393–402.

Pauwels, K., Ambler, T., Clark, B. H., LaPointe, P., Reibstein, D., Skiera, B., Wierenga, B., and Wiesel, T. (2009). Dashboards as a Service: Why, What, How, and What Research Is Needed? *Journal of Service Research*, 12(2):175–189.

Pieters, R., Wedel, M., and Batra, R. (2010). The Stopping Power of Advertising: Measures and Effects of Visual Complexity. *Journal of Marketing*, 74(5):48–60.

Ranganath, R., Gerrish, S., and Blei, D. (2014a). Black box variational inference. In *Artificial Intelligence and Statistics*, pages 814–822.

Ranganath, R., Tang, L., Charlin, L., and Blei, D. M. (2014b). Deep Exponential Families. *arXiv:1411.2581v1*, pages 2–4.

Rasmussen, E. and Williams, K. I. (2006). *Gaussian Processes for Machine Learning*. MIT Press.

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic Backpropagation and Approximate Inference in Deep Generative Models. *Proceedings of the 31st International Conference on Machine Learning*.

Roberts, J. H. and Urban, G. L. (1988). Modeling Multiattribute Utility , Risk , and Belief Dynamics for New Consumer Durable Brand Choice. *Management Science*, 34(2):167 – 185.

Roberts, S., Osborne, M., Ebden, M., Reece, S., Gibson, N., and Aigrain, S. (2013). Gaussian processes for time-series modelling. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, 371(1984):20110550.

Rossi, P. E. (2013). Bayesian Semi-parametric and Non-parametric Methods with Applications to Marketing and Micro-econometrics.

Schlosser, A. E., Rikhi, R. R., and Dagogo-Jack, S. W. (2016). The Ups and downs of visual orientation: The effects of diagonal orientation on product judgment. *Journal of Consumer Psychology*.

Schmittlein, D. C., Morrison, D. G., and Colombo, R. (1987). Counting Your Customers: Who-Are They and What Will They Do Next? *Management Science*, 33(1):1–24.

Schweidel, D. A. and Knox, G. (2013). Incorporating Direct Marketing Activity into Latent Attrition Models. *Marketing Science*, 32(3):471–487.

Seetharaman, P. B. and Chintagunta, P. K. (2003). The Proportional Hazard Model for Purchase Timing: A Comparison of Alternative Specifications. *Journal of Business & Economic Statistics*, 21(3):368–382.

Semin, G. R. and Palma, T. A. (2014). Why the bride wears white: Grounding gender with brightness. *Journal of Consumer Psychology*, 24(2):217–225.

Shively, T. S., Allenby, G. M., and Kohn, R. (2000). A Nonparametric Approach to Identifying Latent Relationships in Hierarchical Models. *Marketing Science*, 19(2):149–162.

Singh, V. P., Hansen, K. T., and Gupta, S. (2005). Modeling Preferences for Common Attributes in Multicategory Brand Choice. *Journal of Marketing Research*, 42(2):195–209.

Spence, C. (2012). Managing sensory expectations concerning products and brands: Capitalizing on the potential of sound and shape symbolism. *Journal of Consumer Psychology*, 22(1):37–54.

Sriram, S., Chintagunta, P. K., and Neelamegham, R. (2006). Effects of Brand Preference, Product Attributes, and Marketing Mix Variables in Technology Product Markets. *Marketing Science*, 25(5):440–456.

Sriram, S. and Kalwani, M. U. (2007). Optimal and Promotion Advertising Budgets Dynamic Markets with Brand Equity as a Variable Mediating. *Management Science*, 53(1):46–60.

Suzuki, M., Nakayama, K., and Matsuo, Y. (2017). Joint Multimodal Learning with Deep Generative Models. In *Proceedings of the 5th International Conference on Learning Representations*.

Train, K. E. (2009). *Discrete choice methods with simulation*. Cambridge university press.

Tran, D., Kucukelbir, A., Dieng, A. B., Rudolph, M., Liang, D., and Blei, D. M. (2016). Edward: A library for probabilistic modeling, inference, and criticism. *arXiv preprint arXiv:1610.09787*.

Valdez, P. and Mehrabian, A. (1994). Effects of color on emotions. *Journal of Experimental Psychology: General*, 123(4):394–409.

van der Lans, R., Cote, J. a., Cole, C. a., Leong, S. M., Smidts, A., Henderson, P. W., Bluemelhuber, C., Bottomley, P. a., Doyle, J. R., Fedorikhin, A., Moorthy, J., Ramaseshan, B., and Schmitt, B. H. (2009). Cross-National Logo Evaluation Analysis: An Individual-Level Approach. *Marketing Science*, 28(5):968–985.

Vedantam, R., Fischer, I., Huang, J., and Murphy, K. (2018). Generative Models of Visually Grounded Imagination. In *Proceedings of the 6th International Conference on Learning Representations*.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning - ICML '08*, pages 1096–1103.

Walsh, M. F., Winterich, K. P., and Mittal, V. (2010). Do logo redesigns help or hurt your brand? The role of brand commitment. *Journal of Product & Brand Management*, 19(2):76–84.

Wang, W., Arora, R., Livescu, K., and Bilmes, J. (2015). On Deep Multi-View Representation Learning. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37.

Wedel, M. and Kannan, P. K. (2016). Marketing Analytics for Data-Rich Environments. *Journal of Marketing*, 80(6):97–121.

Wedel, M. and Zhang, J. (2004). Analyzing Brand Competition Across Subcategories. *Journal of Marketing Research*, 41:448–456.

Wilson, A. G. (2014). *Covariance kernels for fast automatic pattern discovery and extrapolation with Gaussian processes*. PhD thesis, University of Cambridge.

Wu, M. and Goodman, N. (2018). Multimodal Generative Models for Scalable Weakly-Supervised Learning. In *32nd Conference on Neural Information Processing Systems (NIPS*.

Yang, J., Zhu, H., Choi, T., and Cox, D. D. (2016). Smoothing and mean-Covariance estimation of functional data with a bayesian hierarchical model. *Bayesian Analysis*, 11(3):649–670.

Zhang, H. (2004). Inconsistent estimation and asymptotically equal interpolations in model-based geostatistics. *Journal of the American Statistical Association*, 99(465):250–261.